AD-A104 901 NAVAL AIR DEVELOPMENT CENTER WARMINSTER PA AIRCRAFT --ETC F/G 20/4
MATH MODEL FOR NADC SPIN SIMULATOR AND F-14 HIGH ANGLE OF ATTAC--ETC(U)
OCT 80 J M STIFEL

UNCLASSIFIED NADC-80220-60-VOL-1 NL

1 OF 2
AD A
104901

LEVEL

(14)

(12)

REPORT NO: NADC-80220-60 - Vol - 1

VOLUME I

MATH MODEL FOR NADC SPIN SIMULATOR
AND F-14 HIGH ANGLE OF ATTACK
DATA BASE

J. M. Stifel

Aircraft and Crew Systems Technology Directorate
NAVAL AIR DEVELOPMENT CENTER
Warminster, Pa. 18974

October 1980

FINAL REPORT:
AIRTASK NO. A0595941/0012/0241/000003
WORK UNIT NO. 1F0960GA
Approved For Public Release, Distribution Unlimited

Prepared for
NAVAL AIR SYSTEMS COMMAND
Department of the Navy
Washington, D.C. 20361

81 10 2 108

NOTICES

REPORT NUMBERING SYSTEM – The numbering of technical project reports issued by the Naval Air Development Center is arranged for specific identification purposes. Each number consists of the Center acronym, the calendar year in which the number was assigned, the sequence number of the report within the specific calendar year, and the official 2-digit correspondence code of the Command Office or the Functional Directorate responsible for the report. For example: Report No. NADC-78015-20 indicates the fifteeth Center report for the year 1978, and prepared by the Systems Directorate. The numerical codes are as follows:

CODE	OFFICE OR DIRECTORATE
00	Commander, Naval Air Development Center
01	Technical Director, Naval Air Development Center
02	Comptroller
10	Directorate Command Projects
20	Systems Directorate
30	Sensors & Avionics Technology Directorate
40	Communication & Navigation Technology Directorate
50	Software Computer Directorate
60	Aircraft & Crew Systems Technology Directorate
70	Planning Assessment Resources
80	Engineering Support Group

PRODUCT ENDORSEMENT – The discussion or instructions concerning commercial products herein do not constitute an endorsement by the Government nor do they convey or imply the license or right to use such products.

APPROVED BY: J. R. Woods DATE: 7/31/81

J. R. WOODS
CDR USN

AD-A104 901

NAVAL AIR DEVELOPMENT CENTER
AIRCRAFT & CREW SYSTEMS TECHNOLOGY DIRECTORATE
WARMINSTER, PA 18974

6053
20 OCT 1981

MEMORANDUM

From: Mr. J. M. Stifel, Naval Air Development Center (6051)

Subj: Corrections to NAVAIRDEVCEN Report No. NADC-80220-60

Ref: (a) NAVAIRDEVCEN Report No. NADC-80220-60, "Math Model for NAVAIRDEVCEN Spin Simulator and F-14 High Angle of Attack Data Base" Volume I, J. M. Stifel, October 1980

Encl: (1) Errata Sheet for NAVAIRDEVCEN Report No. NADC-80220-60

1. Despite rigorous attempts at editing during the publication cycle, the final version of reference (a) contains several errors which may cause confusion.

2. It is advised that all those who received copies of this report insert a copy of the enclosed errata sheet in each report. Corrections to the printed copy of the reference report may be made at the discretion of the holder.

J. M. STIFEL
Flight Dynamics Branch
Aircraft & Crew Systems Technology Directorate

Distribution:
NAVAIR (AIR-950D, -320D, -5301)
NASA Langley (Mr. W. E. Gilbert)
NASA Dryden (Mr. J. Gera, E-EDC)
ASN/R&D (Mr. L. V. Schmidt)
DTIC

NAVAL AIR DEVELOPMENT CENTER
AIRCRAFT AND CREW SYSTEMS TECHNOLOGY DIRECTORATE
WARMINSTER, PA 18974

6053
19 OCT 1981

ERRATA SHEET
NADC REPORT NO. 80220-60
MATH MODEL FOR NADC SPIN SIMULATOR
AND F-14 HIGH ANGLE OF ATTACK DATA BASE
J. M. STIFEL, OCTOBER 1980
VOLUME I

Page	Comments
11	First line; "is" to "are" 3rd paragraph; "page 12" to "page 10"
21	3rd equation "P_{ij}" to "P_{1j}"
28	Sweep Rate SW "$\Lambda = \pm E_{12}$" to "$\dot{\Lambda} = \geq E_{12}$"
43	First curve mislabeled should be "Directional Gearing" Vertical axis; "δr ~ DEG" Directional spring enclosed in Figure 1
A-5	2nd line of first equation belongs to right of equal sign
A-6	First equation "$m \geq h$" to "$m \geq n$"
A-15	Second line of second equation belongs to right of equals sign
B-1	Third equation; "Am-l" to "Am-i"
B-3	Six lines from bottom; "systems" to "system"
B-5	Center of page "night" to "right"
B-6	Fifth line from bottom "Ax(t)" to "$A\bar{x}(t)$"
B-7	Second equation "Bi" to "B_1" and "Bm" to "B_m"
B-14	Second figure, leftside "Y_2" to "Y_1"

Enclosure (1)

F-14

DIRECTIONAL [illegible]

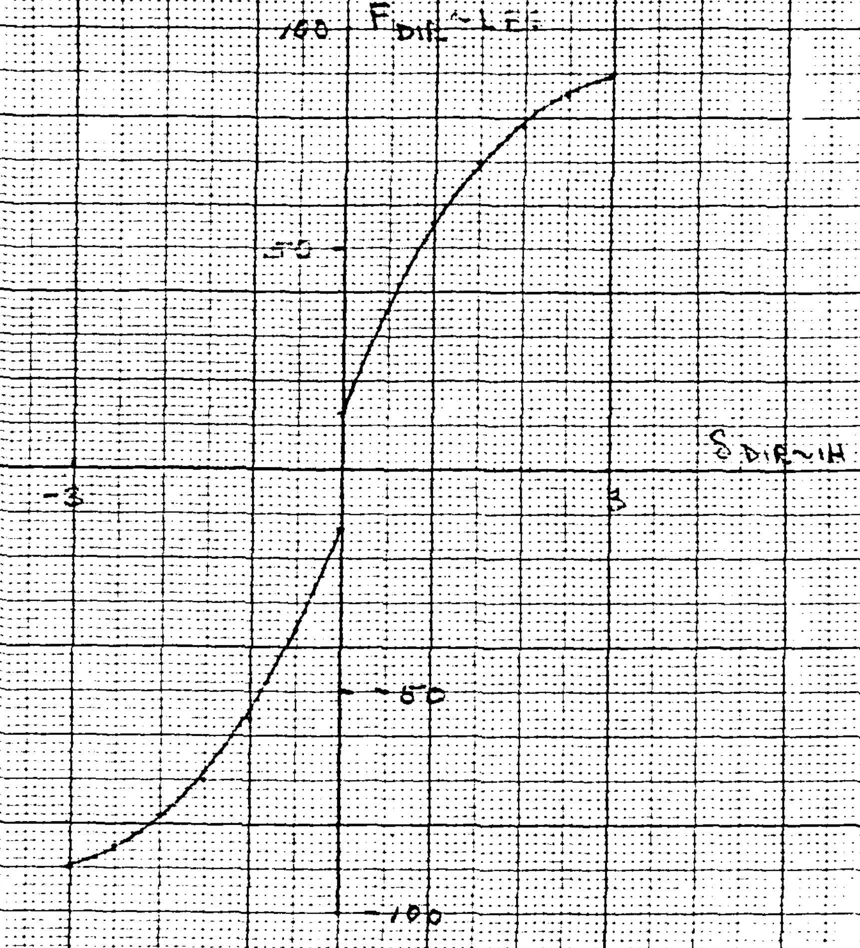


FIGURE 2

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NADC-80220-60	2. GOVT ACCESSION NO. AD-A104901	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) MATH MODEL FOR NADC SPIN SIMULATOR AND F-14 HIGH ANGLE OF ATTACK DATA BASE: VOL. I		5. TYPE OF REPORT & PERIOD COVERED FINAL: JUNE 1981
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) J. M. Stifel		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Aircraft & Crew Systems Technoolgy Directorate Naval Air Development Center Warminster, PA 18974		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Airtask #A0595941/0012/0241/000003 Work Unit No. 1F0960GA
11. CONTROLLING OFFICE NAME AND ADDRESS NAVAL AIR SYSTEMS COMMAND DEPARTMENT OF THE NAVY WASHINGTON, D.C. 20361		12. REPORT DATE October 1980
		13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release, Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

This report consists of 2 Volumes:
Volume I (Unclassified) & Volume II (Confidential)

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

High Angle of Attack
Spin
Simulation
F-14

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Nonlinear equations of aircraft motion, a digital representation of aircraft control systems, and a high angle of attack data base for the F-14 airplane are presented.

The equations of motion are designed to be suitable for modeling extreme high angle of attack flight and even fully developed spins. These equations accommodate: linear or nonlinear data which are functions of up to three variables, large angle

20. (Continued)

orientations, mass-engine-store assymetries, and oscillatory and rotary balance data.

The control system development uses the Tustin transformation to calculate the state space matrix representation of linear networks directly from the transfer functions of each element. Nonlinear elements such as gain scheduling, switching, and nonlinear algebraic functions are also considered. Current F-14 Control Systems of interest are diagrammed in a manner amenable to the programming of this method.

ACKNOWLEDGEMENTS

The co-operation and assistance of the following individuals in obtaining data and background information for the report is gratefully acknowledged.

L. V. Schmidt; Naval Post Graduate School; Monterey, California.

J. Gera, E-EDC, NASA Dryden FRC, Edwards AFB California.

W. E. Gilbert; NASA Langler Research Center; Langley, Virginia.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

Symbol	Description	Dimensions
A	Characteristic matrix in state space notation	-
A_{ij}	Entries of the characteristic matrix	-
$a_1, a_2, \ldots, a_m$	Coefficients of the output derivatives in m^{th} order ordinary differential equation	-
a_x, a_y, a_z	Body axis components of linear acceleration (inertial)	(ft/sec^2)
a	Speed of sound	(ft/sec)
B	Input Matrix in state space notation	-
B_{ij}	Entries of the input matrix	-
$b_o, b_1, \ldots, b_n$	Coefficients of the input derivatives in m^{th} order ordinary differential equation	-
b	Reference wing span	(ft)
C	Output coupling matrix in state space notation	-
C_{ij}	Entries of C matrix	-
$\bar{c}$	Reference mean aerodynamic chord	(ft)
C_x C_y C_z	Body Axis Force coefficients	-
C_l C_m C_n	Body Axis Moment coefficients	-
D	Matrix in state space notation	-
D_{ij}	Entries of D matrix	-
d_{ij}	Entries of the inverse of the inertia matrix	-
det	Determinate of	-
E	Input coupling matrix in state space notation	-
E_{ij}	Entries of E matrix	-

Symbol	Description	Dimensions
$F_1, F_2, \ldots, F_m$	Functions used for defining state variables	-
$\bar{F}$	A column vector comprised of $F_1, \ldots, F_m$	-
F_x, F_y, F_z	Body axis external force components (inertial)	(lb)
f	A general function of time	-
G	A transfer function	-
G_{ij}	An entry in the transfer function matrix	-
g	Gravitational acceleration	(ft/sec^2)
H	A general feedback loop transfer function	-
I	The identity matrix	-
I_n	Inertia Matrix	-
I_x, I_y, I_z	Moments of Inertia	$(slug\text{-}ft^2)$
I_{xy}, I_{yz}, I_{xz}	Products of Inertia	$(slug\text{-}ft^2)$
I_e	Engine shaft inertia	$(slug\text{-}ft^2)$
i, j, k, l, m	As subscripts; general variables that may be defined and/or re-defined in text	-
l	Rolling moment	(ft-lb)
l_{yk}	Distance between xz body axis plane and engine thrust line for k^{th} engine (+ left, - right)	(ft)
l_{Tk}	Perpendicular distance between engine thrust line and nominal cq. position for k^{th} engine. Measured in xz body axis plane.	(ft)
M	Mach number	-
M_x, M_y, M_z	Sum of external applied, inertia coupling, and cg. shift moments in body axes	(ft-lbs)
m	Pitching moment	(ft-lb)
n	Yawing moment	(ft-lb)
$\dot{p}, p$	Roll acceleration, and rate, inertial	(sec^{-2}, sec^{-1})
Q	Dynamic pressure	(lb/ft^2)
$\dot{q}, q$	Pitch acceleration and rate, inertial	(sec^{-2}, sec^{-1})
$\dot{r}, r$	Yaw acceleration and rate, inertial	(sec^{-2}, sec^{-1})

Symbol	Description	Dimensions
S	Reference wing area	(ft^2)
s	Laplace variable	(sec^{-1})
T_k	Thrust of the k^{th} engine	(lb)
$\dot{u}, \dot{v}, \dot{w}$	Body axis acceleration components, (non-inertial)	(ft/sec^2)
u, v, w	Body axis component of inertial velocity	(ft/sec)
u	A general input in state space notation	-
$\overline{u}$	A column vector of all system inputs	-
$u_1, u_2, \ldots u_k$	The components of $\overline{u}$	-
V_T	Total inertial speed	(ft/sec)
W	Total aircraft weight	(lb)
$\dot{W}_{fi}$	Fuel weight flow for the i^{th} engine	(lb/sec)
X	A matrix of all system state variables	-
$\overline{x}_1$	A column vector of the first state variable of each element in the system	-
$\overline{x}$	A column vector of all state variables of a particular element	-
$\overline{y}$	A column vector of all output variables	-
$\vec{\alpha}$	Angular acceleration	(sec^{-2})
α	Angle of attack	(rad)
β	Angle of sideslip	(rad)
$\dot{\alpha}, \dot{\beta}$	First derivatives of α, β	(sec^{-1})
ϵ	Angle in xz body axis plane, between thrust center line and longitudinal axis	(deg)
ζ	General damping ratio	-
σ or $\frac{1}{\tau}$	Real part of Laplace variable	(sec^{-1})
λ	General eigenvalue	-
ψ, θ, ϕ	Standard aircraft Euler Angles	(deg)
$\psi_{-1}, \theta_{-1}, \phi_{-1}$	Inverse Euler Angles	(deg)

Symbol	Description	Dimension
ω	Frequency, imaginary part of Laplace variable	(sec^{-1})
ω_n	Undamped natural frequency	(sec^{-1})
ω_e	Engine rotor angular rate	(sec^{-1})
$\vec{\Omega}$	Angular velocity	(sec^{-1})
$\omega_x, \omega_y, \omega_z$	Body axis components of angular velocity	(sec^{-1})
$\vec{\omega}$	Inertial angular velocity	(sec^{-1})

INTRODUCTION

Crew and aircraft losses attributed to loss of control at high angles of attack continue to occur and to be of concern to both the Navy and Air Force, and continue to be identified as an area in which further research is needed. The need exists not only to reduce such losses of current aircraft but to minimize those of future aircraft now in development.

In 1979, the Navy appointed an independent Executive Review Group to assess the problem relative to the F-14. Among its conclusions the group identified causal factors associated with airframe aerodynamics, control system mechanization, engine asymmetries, piloting technique, and physiological factors. The group also recommended high fidelity simulation as a low risk means of addressing this problem.

Realizing the potential of the NADC centrifuge (dynamic flight simulator) for this purpose, the Aircraft and Crew Systems Technology Directorate has begun a program to substantially improve its simulation capability to meet the requirements of a high fidelity simulator capable of modeling a variety of aircraft. This capability will encompass most conventional simulation techniques, and it will have the capability and flexibility to model aircraft dynamics in all flight conditions including fully developed spins.

As part of that program the Flight Dynamics Branch (Code 6053) was asked to provide equations of motion and F-14 aerodynamic and control system data adequate to the task. This report is intended to document those recommendations and to serve as a basis for subsequent simulation software revisions. Since the intent is to use the simulation as a research tool, considerable emphasis is given to flexibility.

It is considered beyond the scope of this document to implement, in the form of programs and subroutines, the recommendations presented. In the interest of completeness however, certain programming approaches are suggested. While efforts are made to make those suggestions readily programmable, it is realized that the suggested methods may not necessarily be the most computationally efficient. While some changes may be required, it is felt that implementation of all major features can be preserved in the final software implementation.

EQUATIONS OF MOTION

NONLINEARIZED DYNAMIC EQUATIONS

It is the need for the proposed simulation to represent the extreme high angle of attack flight regime which distinguishes it from other simulations. Therefore the requirements of this feature are the driving factors in settling on the equations of motion. Several considerations peculiar to this type of simulation may be listed as:

1. Nonlinearity of data

2. Uncertainty about form of data

3. Importance of various asymmetries

4. Large angle motions

Just how each of these factors impacts the form of the equations of motion will be pointed out in this section.

The aerodynamic force and moment data typically changes very rapidly and often unexpectedly with changes in the orientation of the free stream velocity vector. Furthermore, there are no guarantees that trim is possible at all flight conditions of interest.

The aircraft may undergo such extremely violent excursions that any state of equilibrium may be simply a very brief transient state. Under these conditions the conventional stability derivatives have little meaning. That is, the concept of aircraft motions consisting of small perturbations from an equilibrium condition is not applicable. For this reason the usual linearization of the rigid body dynamic equations, whether for analysis or simulation, is inappropriate. The full nonlinear rigid body dynamic equations must be used.

These equations arise from the six degree of freedom equations of Newtonian dynamics.

$$\vec{F} = \frac{d}{dt} m\vec{V} \qquad \vec{T} = \frac{d}{dt} I_n \vec{\Omega}$$

which define force and moment as the first derivative of linear and angular momentum, respectively. These equations are defined for inertial (nonaccelerating, nonrotating) reference frames.

If we wish to write the force and moment equations in a reference frame that may be rotating, we must remember that differentiation of any vector V in such a system requires an additional term.

$$\frac{d\vec{V}}{dt} = \dot{\vec{V}} + \vec{\omega}_{co\text{-}ord} \times \vec{V}$$

to represent the inertial quantity.

The assumption that mass and inertia are approximately constant for the time required to pass through the equations of motion once yields.

$$\vec{F} = m\vec{a}^{*} \qquad \vec{T} = I_n \dot{\vec{\Omega}} + (\vec{\Omega} \times I_n \vec{\Omega})$$

Making this assumption essentially selects body axes as the reference frame for all forces and moments because it is the only reference frame in which inertias are constant with time. To avoid unnecessary axes conversions, it will be assumed that all aerodynamic data will be in body axes.

If the aircraft motion is considered to be a combination of translation of the center of mass and rotation about the center of mass then all forces may be considered to act through, all mass may be considered concentrated in, and the origin of the body axis system may be attached to, the center of gravity.

By, differentiating the expression for position in body axis twice, using the prescribed differentiation formula, we may arrive at the standard dynamic equation for inertial acceleration.

$$\vec{V} = \frac{d\vec{R}}{dt} = \dot{\vec{R}} + \vec{\Omega} \times \vec{R}$$

$$\vec{a} = \frac{dV}{dt} = \underbrace{(\ddot{\vec{R}} + \dot{\vec{\Omega}} \times \vec{R} + \vec{\Omega} \times \dot{\vec{R}})}_{\dot{\vec{V}}} + \underbrace{\vec{\Omega} \times \dot{\vec{R}} + \vec{\Omega} \times \vec{\Omega} \times \vec{R}}_{\vec{\Omega} \times \vec{V}}$$

Since the center of mass is always at the origin, R = 0, and body axis velocity is true inertial velocity, its components are defined as

$$\vec{V}_T = u\,\vec{i} + v\,\vec{j} + w\,\vec{k}$$

But body axis acceleration is related to inertial acceleration by

$$\vec{a} = \dot{\vec{V}} + (\vec{\Omega} \times \vec{V}).$$

* $\vec{a}$ is defined as inertial acceleration.

Similar arguments apply to show that the body axis components of angular velocity are the components of the true inertial rate.

$$\vec{\Omega} = p\vec{i} + q\vec{j} + r\vec{k}.$$

Angular acceleration in body axis is also the inertial angular acceleration since the coriolus terms ($\vec{\Omega} \times \vec{\Omega}$) are zero by the properties of the vector cross product.

$$\vec{\alpha} = \dot{p}\vec{i} + \dot{q}\vec{j} + \dot{r}\vec{k}.$$

Expanding the linear acceleration equation gives the more common form

$$\dot{u} = a_x + (rv - qw)$$
$$\dot{v} = a_y + (pw - ru)$$
$$\dot{w} = a_z + (qu - pv)$$

Because accelerations can be easily integrated to obtain rates and positions, it is most useful to write the force and moment equations in terms of accelerations.

It is still a matter of choice as to which forces and moments are included in the equations. The significant forces may be categorized as aerodynamic, thrust, and gravitational, making the rectilinear equations

$$a_x = g/W \left\{ QS\Sigma\, C_{xi} + \cos\epsilon\, \Sigma T_i \right\} - g \sin\theta$$
$$a_y = g/W \left\{ QS\Sigma\, C_{yi} \right\} + g \cos\theta \, \sin\phi$$
$$a_z = g/W \left\{ QS\Sigma\, C_{zi} + \sin\epsilon\, \Sigma\, T_i \right\} + g \cos\theta \, \cos\phi.$$

Similarly the moment equation

$$\vec{T} = I_n \dot{\vec{\Omega}} + (\vec{\Omega} \times I_n \vec{\Omega})$$

may be written

$$\dot{\vec{\Omega}} = I_n^{-1} \left\{ \vec{T} - (\vec{\Omega} \times I_n \vec{\Omega}) \right\}$$

or

$$\dot{p} = d_1\, M_x + d_{12}\, M_y + d_{13}\, M_z$$
$$\dot{q} = d_{12}\, M_x + d_2\, M_y + d_{23}\, M_z$$
$$\dot{r} = d_{13}\, M_x + d_{23}\, M_y + d_3\, M_z$$

Where M_x, M_y, M_z, include aerodynamic moments, thrust moments, corrections for center of gravity shift, as well as inertia coupling moments, and engine gyroscopic moments.

If the applied force components are defined as:

$$F_x = Qs \ \Sigma C_{xi} + \cos\epsilon \ \Sigma T_i$$

$$F_y = Qs \ \Sigma C_{yi}$$

$$F_z = Qs \ \Sigma C_{zi} + \sin\epsilon \ \Sigma T_i$$

the form of the moment terms are kept more manageable.

$$\begin{aligned} M_x = {} & QSb \ \Sigma C_{li} + (I_y - I_z) qr + p (qI_{xz} - rI_{xy}) \\ & + (q^2 - r^2) I_{yz} + \sin\epsilon \ \Sigma l_{yi} T_i \\ & + F_y (\Delta z_{cg}) - F_z (\Delta y_{cg}) \\ & + q \sin\epsilon \ \Sigma I_e \omega_{ei} \end{aligned}$$

$$\begin{aligned} M_y = {} & QS\bar{c} \ \Sigma C_{mi} + (I_z - I_x) pr + q (rI_{xy} - pI_{yz}) \\ & + (r^2 - p^2) I_{xz} + \Sigma l_{Ti} T_i \\ & + F_z (\Delta x_{cg}) - Fx (\Delta z_{cg}) \\ & - r \cos\epsilon \ \Sigma I_e \omega_{ei} - p \sin\epsilon \ \Sigma I_e \omega_{ei} \end{aligned}$$

$$\begin{aligned} M_z = {} & QSb \ \Sigma C_{ni} + (I_x - I_y) pq + r (pI_{yz} - qI_{xz}) \\ & + (p^2 - q^2) I_{xy} + \cos\epsilon \ \Sigma l_{yi} T_i \\ & + F_x (\Delta y_{cg}) - F_y (\Delta x_{cg}) + q \cos\epsilon \ \Sigma I_e \omega_{ei} \end{aligned}$$

MASS AND INERTIA ASYMMETRIES

An important factor in the choice of these equations is the ability to account for mass and engine asymmetries.

To this end, the cg. may be shifted in any of three directions from its nominal location (Δx_{cg}, Δy_{cg}, Δz_{cg}), no products of inertia are assumed zero, and each thrust and engine moment arm is accounted for separately.

In order to account for the weight and inertia contribution of individual aircraft or store components, these are expressed as the summation of terms some of which may be functions of variables such as switch positions, wing sweep, or time.

$$I_x = \Sigma I_{xi} \qquad I_{xy} = \Sigma I_{xzi}$$

$$I_y = \Sigma I_{yi} \qquad I_{yz} = \Sigma I_{yzi}$$

$$I_z = \Sigma I_{zi} \qquad I_{xz} = \Sigma I_{xzi}$$

$$W = \Sigma W_i$$

The numbers, d_i, represent the entries of the inverse of the inertia matrix

$$I_n^{-1} = \begin{bmatrix} d_1 & d_{12} & d_{13} \\ d_{12} & d_2 & d_{23} \\ d_{13} & d_{23} & d_3 \end{bmatrix} = \begin{bmatrix} I_x & -I_{xy} & -I_{xz} \\ -I_{xy} & I_y & -I_{yz} \\ -I_{xz} & -I_{yz} & I_z \end{bmatrix}^{-1}$$

Since the product of the inertia matrix and its inverse are the identity matrix.

$$I_n I_n^{-1} = I$$

We can write

$$I_n \begin{bmatrix} d_1 \\ d_{12} \\ d_{13} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

and find each entry by Cramer's rule which for d_1 gives

$$d_1 = \frac{1}{\det I_n} \begin{vmatrix} 1 & -I_{xy} & -I_{xz} \\ 0 & I_y & -I_{yz} \\ 0 & -I_{yz} & I_z \end{vmatrix}$$

In doing so we obtain the relations

$$\det I_n = I_x I_y I_z - 2\, I_{xy} I_{xz} I_{yz} - (I_x I_{yz}^2 + I_y I_{xz}^2 + I_z I_{xy}^2)$$
$$d_1 = (I_y I_z - I_{yz}^2)/\det I_n$$
$$d_2 = (I_x I_z - I_{xz}^2)/\det I_n$$
$$d_3 = (I_x I_y - I_{xy}^2)/\det I_n$$
$$d_{12} = (I_{yz} I_{xz} + I_{xy} I_z)/\det I_n$$
$$d_{13} = (I_{xy} I_{zy} + I_{xz} I_y)/\det I_n$$
$$d_{23} = (I_{xy} I_{xz} - I_{yz} I_x)/\det I_n$$

LARGE ANGLE MOTION CONSIDERATIONS

Since the simulation may be required to travel through very large angular motions it is essential to describe both wind vector and earth axis orientation so that all possible orientations are algebraically definable and continuous.

Velocity vector orientation is definable any time total velocity is not zero. Its magnitude is given by:

$$\left|\vec{V}_T\right| = \sqrt{u^2 + v^2 + w^2}$$

If $V_T \neq 0$, sideslip is defined as

$$\beta = \sin^{-1}\left(\frac{v}{V_T}\right)$$

which is continuous over the range $-90° \leq \beta \leq 90°$. Angle of attack is definable any time the velocity vector has a projection in the XZ body axis plane. That is if

$$u^2 + w^2 \neq 0.$$

The normal definition for angle of attack however,

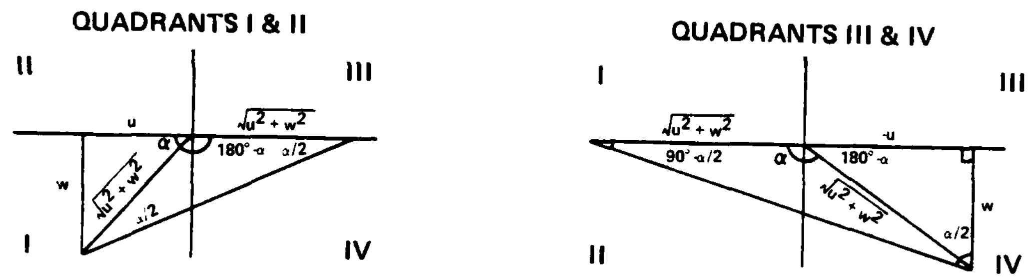
$$\alpha = \tan^{-1} \frac{w}{u}$$

is not defined at $\alpha = \pm 90°$ and further can not distinguish between the range $0° \leq \alpha \leq 90°$ and the range $-90° \leq \alpha \leq 180°$, nor the range $-90° \leq \alpha \leq 0$ and the range $90° \leq \alpha \leq 180°$.

An alternate expression valid over the entire range $-180° < \alpha < 180°$ (not inclusive) is

$$\alpha = 2 \tan^{-1} \left\{ \frac{w}{u + \sqrt{u^2 + w^2}} \right\}$$

For quadrants I and II this expression follows directly from the geometry below.



For quadrants III and IV the geometry says

$$\tan \alpha/2 = \frac{-u + \sqrt{u^2 + w^2}}{w}$$

which can be altered to the desired form by multiplying by

$$\frac{u + \sqrt{u^2 + w^2}}{u + \sqrt{u^2 + w^2}}$$

provided $u + \sqrt{u^2 + w^2} \neq 0$, $\alpha \neq \pm 180°$. Conversely if

$$u = -\sqrt{u^2 + w^2}, \quad \alpha = 180°.$$

Differentiating the expressions defining angle of attack and sideslip and substituting the relations on page 4 between apparent body axis and inertial accelerations gives expressions for rates,

$$\dot{\alpha} = \frac{a_z u - a_x w - v\,(pu + rw)}{(u^2 + w^2)} + q$$

$$\dot{\beta} = \frac{a_y\,(u^2 + w^2) - v\,(a_x u + a_z w) + V_T^2\,(pw - ru)}{V_T^2\,\sqrt{u^2 + w^2}}$$

in terms of readily available quantities.

Similar problems occur in representing earth axis orientation with standard Euler angles which are obtained by integrating the Euler angle rates

$$\dot{\phi} = p + (q \sin\phi + r \cos\phi)\tan\theta$$

$$\dot{\psi} = (q \sin\phi + r \cos\phi)/\cos\theta$$

$$\dot{\theta} = q \cos\phi - r \sin\phi.$$

At $\theta = \pm 90°$ the expressions for $\dot{\phi}$ and $\dot{\psi}$ become undefined and their values ambiguous.

One possible way of avoiding this problem is by use of quaternions which are both continuous and unambiguous. Since this method was previously used for the ACM simulation no elaboration is necessary here.

Should use of an alternate method become necessary, one is suggested by the fact that if the Euler rates are known, no such problem exists.

$$p = \dot{\phi} - \dot{\psi} \sin\theta$$

$$q = \dot{\theta} \cos\theta + \dot{\psi} \cos\theta \sin\psi$$

$$r = \dot{\psi} \cos\theta \cos\phi - \dot{\theta} \sin\phi$$

This directional quality suggests that a set of inverse Euler angles be used to determine the orientation of earth axes relative to body axes. The inverse Euler angles

are defined in the same manner as their conventional counterparts, except that the sequence of rotations begins at body axes and ends with earth axes rather than vice versa.

$$\dot{\psi}_{-1} = r$$

$$\dot{\theta}_{-1} = q \cos \psi_{-1} - p \sin \psi_{-1}$$

$$\dot{\phi}_{-1} = (p \cos \psi_{-1} + q \sin \psi_{-1}) \cos \theta_{-1} - r \sin \theta_{-1}$$

This equation is not singular for any orientation. Furthermore, the matrix which maps to earth axes using inverse Euler angles is identical in form to that which maps earth axes to body axes using the standard set.

$$\begin{bmatrix} \cos\psi \cos\theta & \sin\psi \cos\theta & -\sin\theta \\ \begin{pmatrix} \cos\psi \sin\theta \sin\phi \\ -\cos\phi \sin\psi \end{pmatrix} & \begin{pmatrix} \sin\psi \sin\theta \sin\phi \\ +\cos\psi \cos\phi \end{pmatrix} & \cos\theta \sin\phi \\ \begin{pmatrix} \cos\psi \sin\theta \cos\phi \\ +\sin\psi \sin\phi \end{pmatrix} & \begin{pmatrix} \sin\psi \sin\theta \cos\phi \\ -\cos\psi \sin\phi \end{pmatrix} & \cos\theta \cos\phi \end{bmatrix}$$

Since each transformation is between orthogonal coordinate systems, the inverse of the above matrix is its transpose. Because the above transformation and the identical transformation with the inverse Euler angles substituted are inverse operations, the two sets of angles may be related by equating forms across the diagonal of the above matrix.

$$\theta = -\sin^{-1} \left\{ \cos\psi_{-1} \sin\theta_{-1} \cos\phi_{-1} + \sin\psi_{-1} \sin\phi_{-1} \right\}$$

$$\psi = \cos^{-1} \left\{ \frac{\cos\psi_{-1} \cos\theta_{-1}}{\cos\theta} \right\}$$

$$\phi = \cos^{-1} \left\{ \frac{\cos\theta_{-1} \cos\phi_{-1}}{\cos\theta} \right\}$$

To express the inverse angles in terms of the standard angles simply reverse their positions in this form.

While the expressions for the standard Euler angles is still undefined at $\theta = 90°$, the inverse angles do not experience this problem.

Rates and positions may be calculated by numerical integration. An easy method of implementation is to model each integrator as an element of a linear system using the digital representations to be described later. Those variables needing integration are;

$$u = \int \dot{u} dt \qquad p = \int \dot{p} dt$$

$$v = \int \dot{v} dt \qquad q = \int \dot{q} dt$$

$$w = \int \dot{w} dt \qquad r = \int \dot{r} dt$$

If alternate Euler angles are used to represent earth axis orientation we have also

$$\psi_{-1} = \int \dot{\psi}_{-1} dt$$

$$\theta_{-1} = \int \dot{\theta}_{-1} dt$$

$$\phi_{-1} = \int \dot{\phi}_{-1} dt$$

Standard Euler angles may be determined by direct conversion. Translational rates may be converted to earth axes using the matrix on page 12, and then integrated to keep track of position and altitude

$$X = \int u_e dt$$

$$Y = \int v_e dt$$

$$h = \int w_e dt$$

Using a table look-up to determine Q/m^2 and speed of sound, a, from atmospheric data allows the calculation of

$$M = V_T/a$$

$$Q = (Q/M^2)\, M^2$$

DATA FORM AND USE

An important fact concerning extreme high angle of attack and spin simulation is that the aerodynamic phenomena which generate the forces and moments are not easily predictable or, in some cases, well understood. The significant phenomena may change radically from one aircraft to another, or from one angle of attack or rotation rate range to another. In addition, what phenomena are known to exist on any given aircraft is something which changes with time as new test techniques are developed or as new information is extracted from flight test data.

By providing data storage and equations of motion, we are faced with a great deal of uncertainty about exactly which aerodynamic coefficients are significant, to what extent they may be linear or nonlinear, and even of what variables they may be functions. Essentially what is needed is a programming scheme flexible enough to accept a wide variety of data forms without reprogramming.

A nonlinearized form of the equations of motion have been chosen so that in the absence of linearizing assumptions, either linear or nonlinear data may be used. The aerodynamic data has further been resolved into body axis components and expressed in standard force and moment coefficient form. Each component is expressed simply as a term in a summation. This implies the generalization of data look up process which frees the aerodynamic and thrust data of any specified form.

Under such a programming approach each aerodynamic, thrust, and some weight and inertia terms, would be represented by a data table of up to three dimensions. The table would contain all ordinates and their corresponding data points, but it would also contain a number of control integers. These integers would specify a number of operations to be performed on the result of the raw table look up function.

Such integers would perform the following functions:

1. Specify the number of ordinates in the table

2. Identify each ordinate of the table

3. Identify the equation in which the table represents a term

4. Specify any variables by which the result of the table look up must be multiplied before adding to the equation of motion (for linear data).

To accomplish this it will probably be necessary to establish a coding system whereby any variable (such as accelerations, rates, positions, load factors, mach number altitude, dynamic pressure, etc.) or equation is assigned an integer, which identifies it to the program and allows it to be specified by entering its number in the input data

set. This coding will also be necessary for specification of input and output variables for the control system. It may even be possible, and probably preferable, to use Alpha numeric character sets rather than numerical codes.

It is possible that any data table may be applicable only over a specified range of a particular variable. For instance, a given aerodynamic coefficient may be highly linear over a given range of ordinates and nonlinear outside that range. Therefore, it is necessary to be able to specify a number of variables and limiting values outside of which the table is bypassed and "cut off". Such cut-off variables may not always be restricted to those associated with the table ordinates.

When these features are implemented, it will be possible to add or delete virtually any linear or nonlinear data. The capability exists to completely reprogram the form of the aerodynamic data simply by changing data tables. This capability is essential to the flexibility required to maintain the usefulness of the simulation software in an uncertain and rapidly changing technology area.

EQUATION SUMMARY

Linear Degrees of Freedom:

$$a_x = g\,(F_x/W - \sin\theta)$$

$$a_y = g\,(F_y/W + \sin\phi\,\cos\theta)$$

$$a_z = g\,(F_z/W + \cos\phi\,\cos\theta)$$

Rotational Degrees of Freedom:

$$\dot{p} = d_1\,M_x + d_{12}\,M_y + d_{13}\,M_z$$

$$\dot{q} = d_{12}\,M_x + d_2\,M_y + d_{23}\,M_z$$

$$\dot{r} = d_{13}\,M_x + d_{23}\,M_y + d_3\,M_z$$

Body Axis Force:

$$F_x = QS\,\Sigma\,C_{xi} + \cos\epsilon\;\Sigma\,T_i$$

$$F_y = QS\,\Sigma\,C_{yi}$$

$$F_z = QS\,\Sigma\,C_{zi} + \sin\epsilon\;\Sigma\,T_i$$

Body Axis Moment:

$$M_x = QSb\,\Sigma\, C_{\ell i} + \sin\,\Sigma\, \ell_{yi}\, T_i$$

$$+ F_y\,(\Delta z_{cg}) - F_z\,(\Delta y_{cg}) + q\,\sin\epsilon\,\Sigma\, I_e \omega_{ei}$$

$$+ (I_y - I_z)qr + p(qI_{xz} - rI_{xy}) + (q^2 - r^2)I_{yz}$$

$$M_y = QS\bar{c}\,\Sigma\, C_{mi} + \Sigma\ell_{Ti}\, T_i$$

$$+ F_z\,(\Delta x_{cg}) - F_x\,(\Delta z_{cg}) - (r\cos\epsilon + p\sin\epsilon)\,\Sigma\, I_e \omega_{ei}$$

$$+ (I_z - I_x)\,pr + q\,(r\,I_{xy} - pI_{yz}) + (r^2 - p^2)\,I_{xz}$$

$$M_z = QSb\,\Sigma\, C_{ni} + \cos\epsilon\,\Sigma\ell_{yi}\, T_i$$

$$+ F_x\,(\Delta y_{cg}) - F_y\,(\Delta x_{cg}) + q\,\cos\epsilon\,\Sigma\, I_e \omega_{ei}$$

$$+ (I_x - I_y)\,pq + r\,(pI_{yz} - qI_{xz}) + (p^2 - q^2)\,I_{xy}$$

Weight and Inertia:

$$I_x = \Sigma I_{xi} \qquad I_y = \Sigma I_{yi} \qquad I_z = \Sigma I_{zi}$$

$$I_{xy} = \Sigma I_{xyi} \qquad I_{yz} = \Sigma I_{yzi} \qquad I_{xz} = \Sigma I_{xzi}$$

$$W = \Sigma\, W_i$$

$$\det I_n = I_x I_y I_z - 2I_{xy} I_{xz} I_{yz} - (I_x I_{yz}^2 + I_y I_{xz}^2 + I_z I_{xy}^2)$$

$$d_1 = (I_y I_z - I_{yz}^2)/\det I_n$$

$$d_2 = (I_x I_z - I_{xz}^2)/\det I_n$$

$$d_3 = (I_x I_y - I_{xy}^2)/\det I_n$$

$$d_{12} = (I_{yz} I_{xz} + I_{xy} I_z)/\det I_n$$

$$d_{13} = (I_{xy} I_{zy} + I_{xz} I_y)/\det I_n$$

$$d_{23} = (I_{xy} I_{xz} + I_{yz} I_x)/\det I_n$$

Velocity Vector Orientation:

$$V_T = \sqrt{u^2 + v^2 + w^2}$$

IF $V_T = 0, \quad \beta = \beta_{LAST}$

IF $V_T \neq 0$

$$\beta = \sin^{-1}\left(\frac{v}{V_T}\right)$$

IF $u^2 + w^2 = 0, \quad \alpha = \alpha_{LAST}, \quad \dot{\alpha} = \dot{\alpha}_{LAST}, \quad \dot{\beta} = \dot{\beta}_{LAST}$

IF $u = -\sqrt{u^2 + w^2}\,, \quad \alpha = 180°$

IF $u \neq -\sqrt{u^2 + w^2}$

$$\alpha = 2 \tan^{-1}\left\{\frac{w}{u + \sqrt{u^2 + w^2)}}\right\}$$

$$\dot{\alpha} = \frac{a_z u - a_x w - v\,(pu+rw)}{(u^2 + w^2)} + q$$

$$\dot{\beta} = \frac{a_y (u^2 + w^2) - v\,(a_x u + a_z w) + V_T^2\,(pw-ru)}{V_T^2 \sqrt{u^2 + w^2}}$$

Earth Axis Orientations by Quaternions

Body axis rate derivatives

$$\dot{u} = a_x + (rv - qw)$$

$$\dot{v} = a_y + (pw - ru)$$

$$\dot{w} = a_z + (qu - pv)$$

Obtain by integration

$$u,\ v,\ w,\ p,\ q,\ r,\ \psi,\ \theta,\ \phi$$

$$X,\ Y,\ h$$

Atmosphere dependent quantities

$$M = V_T/a$$

$$Q = (Q/M^2)\ M^2$$

CONTROL SYSTEM DIGITIZATION

APPROACH

Previous simulations on the NAVAIRDEVCEN centrifuge employed an analog representation of the aircraft flight control system. For reasons of run efficiency, reliability, and repeatability, and for the capability to rapidly change control systems as well as to transfer the simulation to other Navy simulation facilities, a digital representation of the aircraft flight control systems is required. The following development outlines a flexible approach for representing such systems on NAVAIRDEVCEN simulation facilities.

Inputs to the control system may be considered to be any of a number of control deflections from various cockpit devices and any of a number of parameters which define the flight condition of the aircraft. Outputs may be surface deflections, thrust level, or stick force feedback terms. The output terms may be used either for determining forces and moments acting on the aircraft or for the operation of cockpit hardware.

All elements of the system may be classified as either linear or nonlinear. A linear element may be defined for this purpose as one described by a transfer function.

Nonlinear elements are those described by nonlinear functions in which time is not an explicit variable, i.e., nonlinear differential equations are not considered. In general, any control system may be viewed as one or more networks of linear elements, separated by one or more nonlinear elements and joined together in a particular element structure.

A general control system may have any number of arbitrarily connected elements. Therefore, it would be extremely difficult to devise a completely general method to model all possible control systems.

It is possible to represent networks of linear elements in a very compact and general fashion using state space notation, and to classify most nonlinear elements as one of a few types. The nonlinear element structure is almost always peculiar to a particular control system and, if altered, will unavoidably require some degree of reprogramming. By altering numerical values which are obtained from input data tables, significant changes may be made without reprogramming.

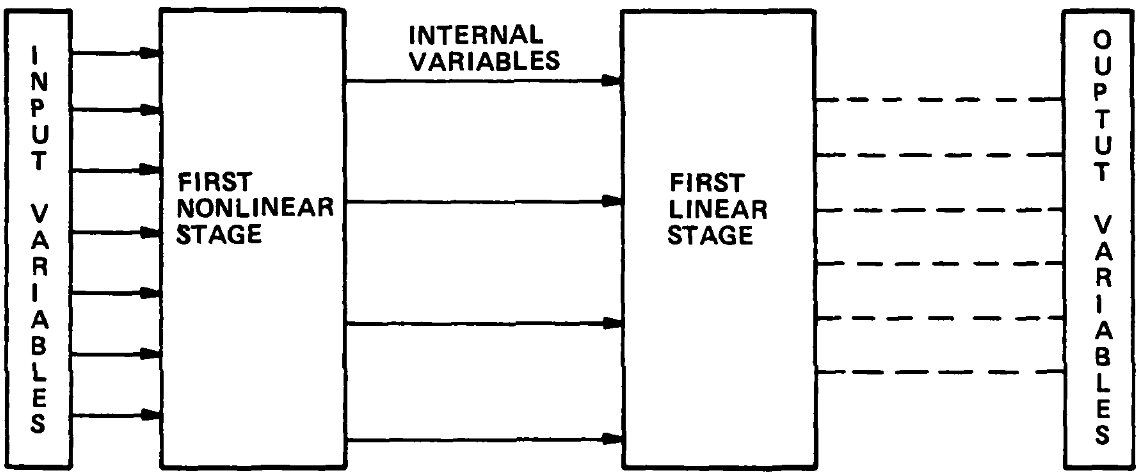
In addition, many different nonlinear element structures may be stored as subroutines, which may be conditionally called, allowing rapid change of control systems.

```
If (ICS. EQ. 1) CALL CONSYS 1 . . . . . . . . . . . . etc.
```

The control system may then be conveniently partitioned into successive nonlinear and linear stages.

The boundaries of a nonlinear stage are established by encountering either a linear element, or the output of a linear element not contained in any previous linear stage, while proceeding in the direction of signal flow.

It is necessary to preserve several variables used internally to the control system to link linear and nonlinear stages. The control system will not generally be expressible as a single linear system. This type of reduction is usually performed assuming all nonlinear test functions are positive.



Most nonlinear elements may be categorized as one of three types.

Schedule	- Any nonlinear function requiring data interpolation or data look up.
Computational Nonlinearity	- A nonlinear function expressible algebraically.
Switching Nonlinearity	- Any logical test including on-off and limiting.

A schedule may represent an element such as a nonlinear gearing, or it may represent programmed configuration changes such as wing sweep, or it may be used to actively change control system parameters: limits, transfer function gains, or coefficients of computational nonlinearities.

Any linear stage may be considered a network of linear elements represented by a transfer matrix equation.

$$\begin{bmatrix} Y_1(s) \\ Y_2(s) \\ \cdot \\ \cdot \\ \cdot \\ Y_j(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) & \cdots & G_{1k}(s) \\ G_{21}(s) & G_{22}(s) & \cdots & G_{2k}(s) \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ G_{j1}(s) & G_{j2}(s) & & G_{jk}(s) \end{bmatrix} \begin{bmatrix} U_1(s) \\ U_2(s) \\ \cdot \\ \cdot \\ \cdot \\ U_k(s) \end{bmatrix}$$

Use of the state space notation to calculate the system outputs minimizes computational requirements.

$$\overline{Y}(t) = CX(t) + D\overline{u}(t)$$

$$X(t + T_s) = A^*X(t) + B\overline{u}(t)$$

This state space form implies both a digital representation of the entries in the transfer function matrix, as well as conversion to a more compact form. Appendix A provides details of how to digitally represent continuous elements, while appendix B explains how to apply state space notation to achieve the above form. The salient features of appendices A and B will be summarized here.

The outputs of each stage may be efficiently calculated in real time knowing only the entries of the two state space matrices C and D and the input and state vectors.

The state vector may be updated between successive calculations of the output vector if the A* and B matrices are known. The entries of the state space matrices may be determined directly from the entries of the transfer function matrix by specifying the following.

For each linear stage:

1. No. of nonzero transfer matrix entries.
2. Order of highest order entry.
3. Number of outputs.
4. Number of inputs.

For each transfer function:

1. Number of first order numerator factors.
2. Number of second order numerator factors.

3. Number of first order denominator factors.

4. Number of second order denominator factors.

5. Transfer function gain (const.).

6. First order time constants, second order frequencies and damping ratios.

7. Location in the transfer matrix (subscripts).

Each transfer function is first expanded into two polynomials in S by an iterative procedure which calculates the coefficients of any polynomial expressed in factored form. If ℓ is temporarily redefined as the number of factors which have been multiplied out, the polynomial after ℓ terms is

$$P = \sum_{i=o}^{\ell} a_{\ell i} \, s^{i}$$

where

$$a_{oo} \doteq 1$$

All complex factors must appear in conjugate pairs to assure all $a_{\ell i}$ are real. Therefore, if the next factor is first order

$$a_{(\ell+1)i} = (a_{\ell\,(i-1)} + a_{\ell i}\,\sigma), \quad \sigma = 1/\tau$$

while noting

$$a_{\ell i} = 0 \text{ if } i < o \quad \text{ or } i > \ell$$

If the next term is second order,

$$a_{(\ell+2)\,i} = (a_{\ell i}\,\omega_n^2 + 2\,\zeta\,\omega_n\,a_{\ell(i-1)} + a_{\ell\,(i-2)})$$

The expansion process is continued until all n numerator coefficients, b_{ni} and all denominator factors, a_{mi}, are obtained.

A matrix of coefficients reflecting the expansion of the first order factors

$$\sum_{j=o}^{m} h_{ij}\, z^{j} = (z-1)^{i}\,(z+1)^{m-i}$$

For $m \geq i \geq o$ comes about as a result of using the Tustin transformation.

$$S = \frac{2}{T_s} \frac{(z-1)}{(z+1)}$$

To produce a difference equation

$$Y(t) = \sum_{j=o}^{m} k \frac{P_{ij}}{P_{2m}} U(t-(m-j)\,T_s) - \sum_{j=o}^{m-1} \frac{P_{2j}}{P_{2m}} Y(t-(m-j)T_s)$$

Whose coefficients

$$P_{ij} = \sum_{i=o}^{n} b_{ni}\, 2^i\, T_s^{m-i}\, h_{ij}$$

$$P_{2j} = \sum_{i-o}^{m} a_{mi}\, 2^i\, T_s^{m-i}\, h_{ij}$$

have the relationships

$$a_i = \frac{P_{2\,m-i}}{P_{2m}}$$

$$b_i = k \frac{P_{1\,m-i}}{P_{2m}}$$

With the state space matrices.

$$A = \begin{bmatrix} -a_1^* & 1 & 0 \cdots 0 \\ -a_2^* & 0 & 1 \cdots 0 \\ \cdot & \cdot & \ddots \\ \cdot & \cdot & 1 \\ \cdot & \cdot & \\ -a_m^* & 0 & 0 \cdots 0 \end{bmatrix}$$

Where $a^*_j = a_{ji}$ is selected from the ith row of the matrix

$$\begin{bmatrix} a_{11} & a_{21} \ldots a_{m1} \\ a_{12} & a_{22} \ldots a_{m2} \\ \vdots & \\ a_{1\ell} & a_{2\ell} \ldots a_{m\ell} \end{bmatrix}$$

one row for each of the ℓ elements.

When multiplying the jth row of A* by the ith column of the state matrix.

$$X = \begin{bmatrix} X_{11} & X_{12} \ldots & X_{1\ell} \\ X_{21} & X_{22} \ldots & X_{2\ell} \\ \vdots & & \\ X_{m1} & X_{m2} \ldots & X_{m\ell} \end{bmatrix}$$

Each column of X is the state vector for each of the ℓ elements in the system.

Also

$$B' = \begin{bmatrix} B_{11} & B_{12} \ldots & B_{1\ell} \\ B_{21} & B_{22} \ldots & B_{2\ell} \\ \vdots & & \vdots \\ B_{m1} & B_{m22} \ldots & B_{m\ell} \end{bmatrix}$$

Where for each of the ℓ elements each column is defined by.

$$B_i = b_i - a_i b_o \qquad m > i \geq 1$$

The matrices C and E are found such that if the ℓth element occupies the jth row and kth column in the transfer matrix

And if $G_\ell = G_{jk}$, $E_{k\ell} = 1$ otherwise $E_{i\ell} = o$

$$C_{\ell j} = 1 \qquad\qquad C_{\ell i} = o.$$

And the matrices B and D are obtained by matrix multiplication

$$B = B' E \text{ and } D = C\, b_o E$$

Where

$$b_o = \begin{bmatrix} b_{o1} & & & O \\ & b_{o2} & & \\ & & \ddots & \\ O & & & b_{o\ell} \end{bmatrix}$$

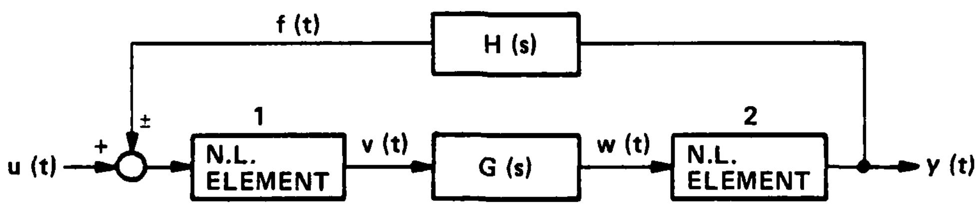
All linear stage schematics may be reduced, through block diagram algebra, to a transfer matrix form.

Only the multiplication of the state space matrices must be accomplished in real time. The matrix entries may be calculated during program initialization. The state matrix should normally also be initially zero.

Some special problems exist with certain types of nonlinearities. In addition, the control system equations may be used to perform certain auxiliary functions such as integration, filtering, or engine dynamics modeling.

If gain scheduling is used on any transfer function, simply set the element gain to unity and calculate the new stage input k(t) U(t) in the previous nonlinear stage.

Closed loop systems containing nonlinear elements present a special problem.



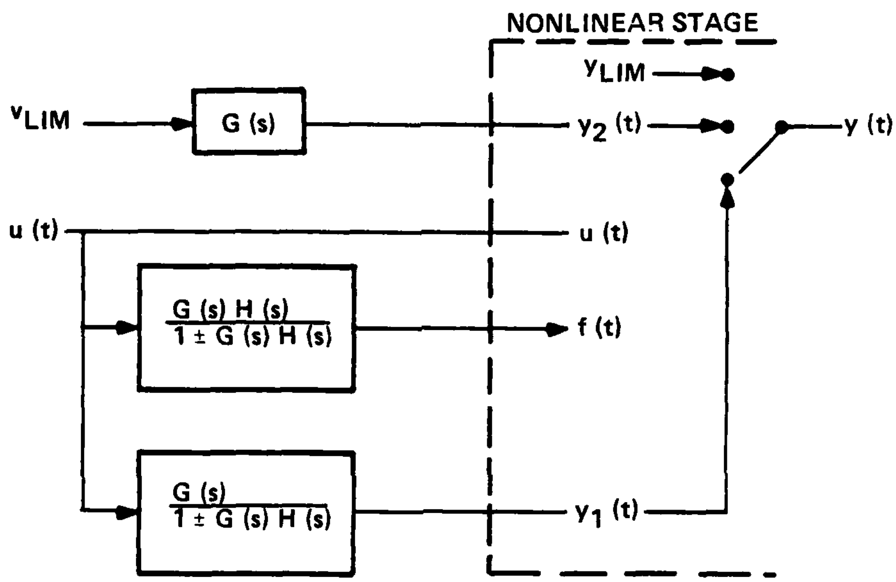
The existence of nonlinear elements inside the feedback loop prevents reduction of the system to a single linear element. Since both u(t) and f(t) must be known to determine y(t); and since y(t) must be known to determine f(t), y(t) must be known to determine y(t) without such a reduction. If the nonlinear elements are schedules or computational nonlinearities, they change the system characteristic equation and necessitate the calculation of the state space matrices in real time. If, however, they are switching functions there are only three possibilities:

1. Switch 2 is open ($y(t) = Y_{lim}$)

2. Switch 2 is closed, switch 1 is open ($v(t) = V_{lim}$)

3. Both switches closed (nonlinearities ignorable).

This system may be represented by three separate systems over the next cycle interval.



The succeeding nonlinear stage may compare $U(t) \pm f(t)$ and V_{lim} to determine if $Y_1(t)$ or $Y_2(t)$ will be compared to Y_{lim}, thus choosing the correct value for $Y(t)$ from among Y_{lim}, $Y_1(t)$, and $Y_2(t)$.

It may be noted that very simple elements, such as multiplication by a constant, may be considered either a zeroth order transfer function or a simple computational nonlinearity.

Because of the time required to execute the logic of the linear element representation, such simple elements will be considered most efficiently represented as computational nonlinearities.

To use rotary and oscillatory balance data for departure and spin modeling, it is necessary to separate the steady and unsteady components of angular rate. In addition, a number of variables describing the aircraft flight condition must be integrated. While these operations are not explicitly shown as part of the control system,

they may be easily represented as a network of linear elements just as are the linear stages, i.e., the subroutine that handles the linear stages may be called from the main program to perform these functions.

Reference 1 contains several F-14 control systems of interest which have been combined here to form control systems A and B. System A represents the current F-14 control system including a Grumman designed Aileron Rudder Interconnect (ARI) intended to alleviate wing rock and roll reversal above 15-degrees angle of attack. The high gain of this system has been found to create lateral pilot induced oscillation (PIO) tendencies above 20-degree angle of attack. For this reason the F-14 is currently being operated with the ARI (Lateral Stability Augmentation System (SAS) switch) off. System B represents a design effort by NASA Langley to alleviate the PIO tendencies of system A with the lateral SAS on, and has recently been flight tested at NASA Dryden.

The following diagrams are based on information taken from references 1 and 2, but they have been rearranged tc facilitate application of the digital control system representation presented here. The engine model is a simple first order filter as used in reference 1. No attempt is made here to model the Approach Power Compensation System.

CONTROL SYSTEM VARIABLES

SYMBOL	DEFINITION	UNITS
SYSTEM INPUTS:		
h	Altitude	ft.
M	Mach No.	—
N_z	Normal Load Factor	(a_z/g)
N_y	Lateral Load Factor	(a_y/g)
p	Roll rate	Sec^{-1}
Q	Dynamic pressure	Lb/Ft^2
$\dot{q}$	Pitch acceleration	Sec^{-2}
q	Pitch rate	Sec^{-1}
r	Yaw rate	Sec^{-1}
α	Angle of attack	Deg.
δ_{DIR}	Rudder pedal deflection	In.
δ_{LAT}	Lateral stick deflection	In.
δ_{LONG}	Longitudinal stick deflection	In.
δ_{SASWD}	Directional SAS switch position	—
δ_{SASWL}	Longitudinal SAS switch position	—
δ_{SASWLT}	Lateral SAS switch position	—
δ_{TRD}	Directional trim switch position	—
δ_{TRL}	Longitudinal trim switch position	—
δ_{TRLT}	Lateral trim switch position	—
δ_{THL}	Left engine throttle position	In.
δ_{THR}	Right engine throttle position	In.

CONTROL SYSTEM VARIABLES (Continued)

SYMBOL	DEFINITION	UNITS
SYSTEM INPUTS (Cont'd.)		
$\delta_{\Lambda man}$	Position of manual sweep override switch	—
Λman	Manually selected sweep angle	Deg.
Λ	Wing sweep angle	Deg.
Λ_{aut}	Output of wing sweep schedule	Deg.
Ω	Total angular rate	Sec^{-1}
SYSTEM OUTPUTS		
$F_{lat.}$	Total force applied to lateral stick	Lb.
F_{long}	Total force applied to longitudinal stick	Lb.
F_{ped}	Total force applied to rudder pedals	Lb.
$\delta_{a\ lim}$	Differential tail deflection (limits applied)	Deg.
$\delta_{e\ lim}$	Stabilator deflection (limits applied)	Deg.
$\delta_{r\ lim}$	Rudder deflection (limits applied)	Deg.
δ_{splim}	Spoiler deflection (limits applied)	Deg.
δ_{tcl}, δ_{tcr}.	Engine thrust levels	% max thrust

All other variables are used internally.

Variables E_{ij} and D_{ij} represent input data associated with switching function and computational nonlinearities in the ith nonlinear stage.

CONTROL SYSTEM DIAGRAMS

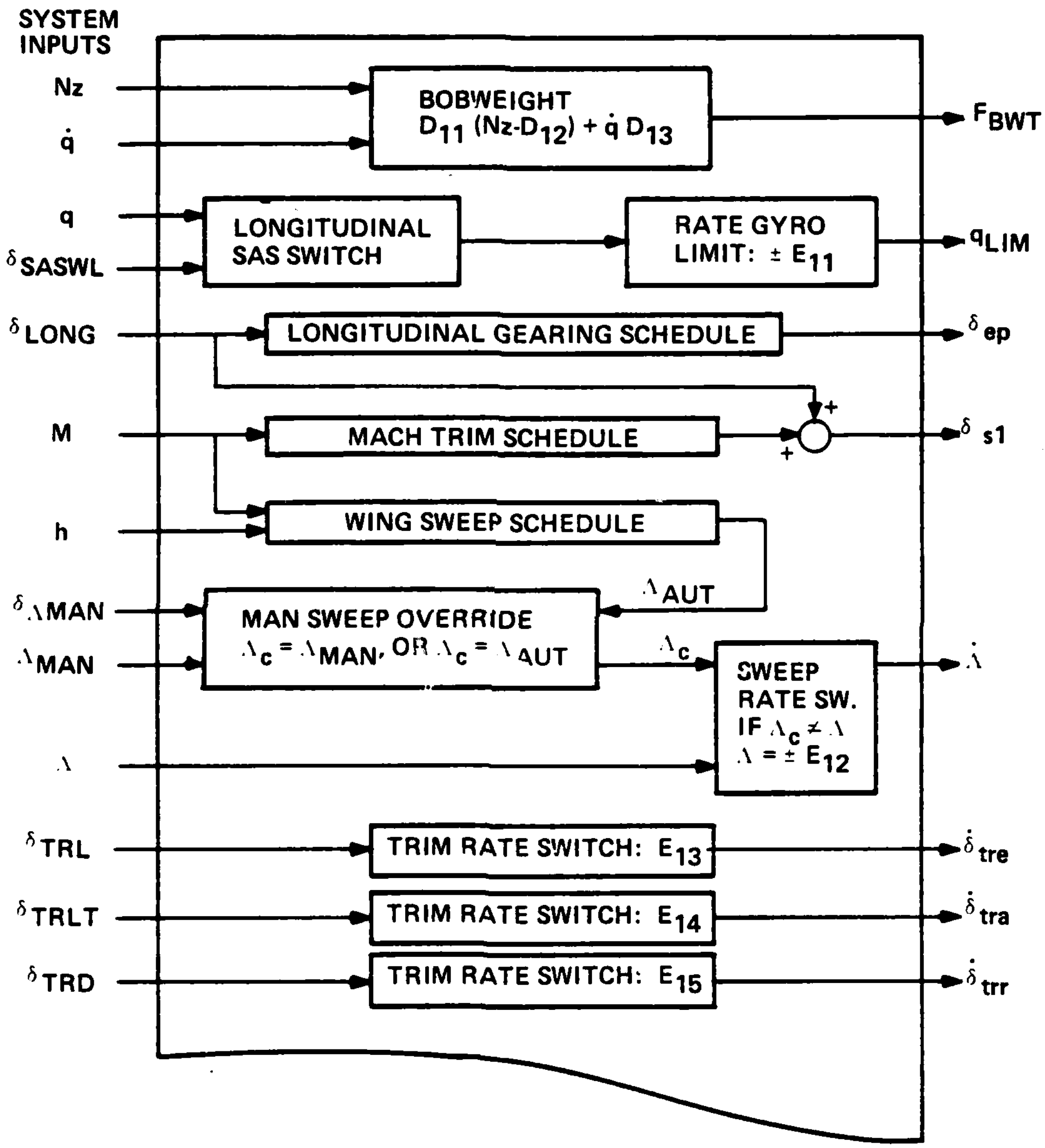


FIGURE 1a. Control System A: First Nonlinear Stage

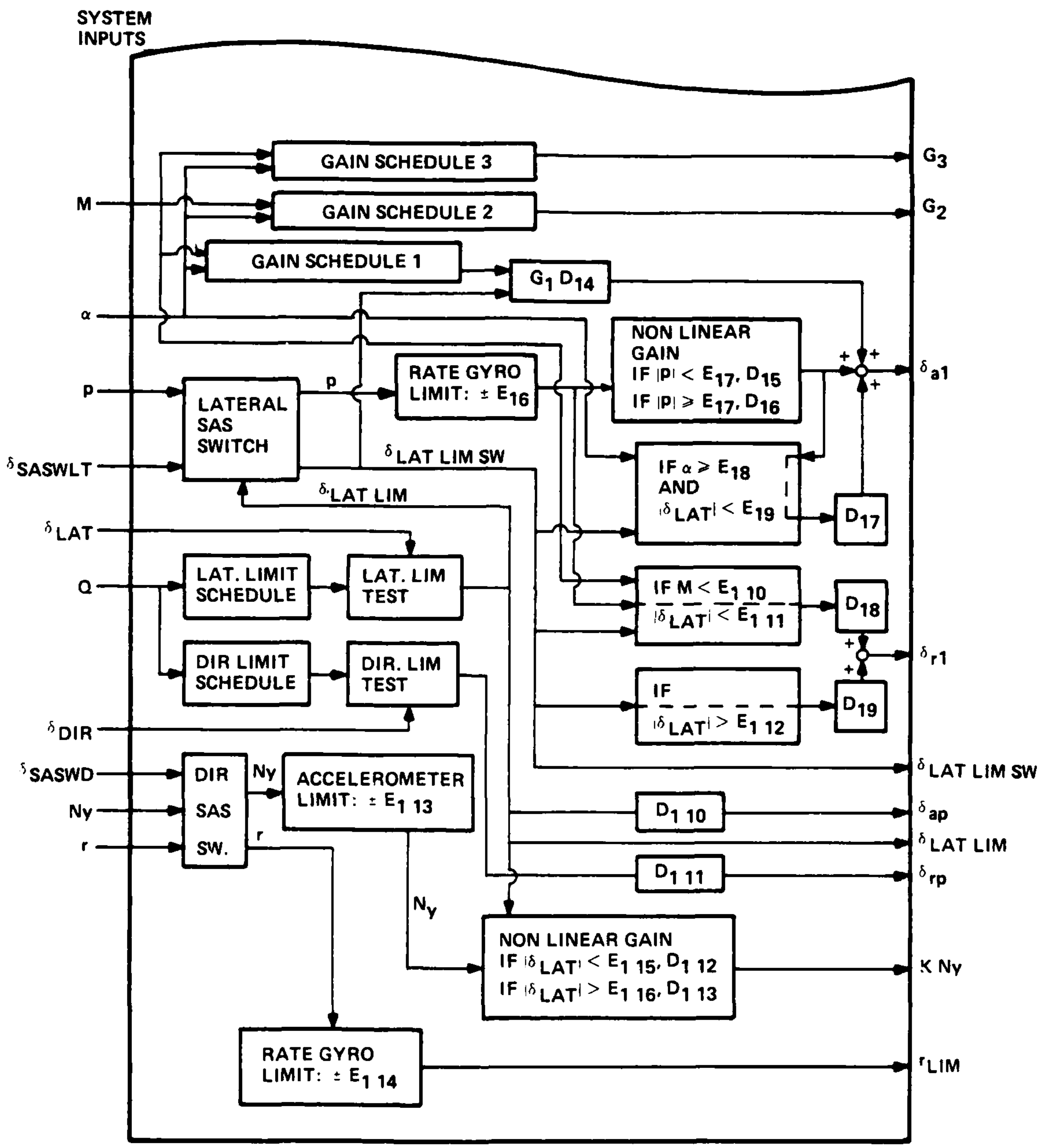


FIGURE 1b. Control System A: First Nonlinear Stage (Cont.)

TABLE I.

CONTROL SYSTEM A: FIRST LINEAR STAGE

FIRST LINEAR STAGE

INPUTS:

U_1	U_2	U_3	U_4	U_5	U_6	U_7	U_8	U_9
q_{LIM}	δr_2	δ LAT LIM SW	K Ny	r_{LIM}	$\dot{\delta}_{tre}$	$\dot{\delta}_{tra}$	$\dot{\delta}_{trr}$	$\dot{\Lambda}_{MAN}$

OUTPUTS:

y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9
δeq_1	δr_2	δa_2	δr_3	δr_4	δ_{tre}	δ_{tra}	δ_{trr}	Λ_{MAN}

Transfer Matrix entries:

$$G_{11} = \frac{S}{S+.5} \qquad G_{22} = \frac{8.}{S+8.} \qquad G_{33} = \frac{8.56}{S+2.}$$

$$G_{44} = \frac{20.}{S+20.} \qquad G_{55} = \frac{S}{S+.5} \qquad G_{66} = \frac{1}{S}$$

$$G_{77} = \frac{1}{S} \qquad G_{88} = \frac{1}{S} \qquad G_{99} = \frac{1}{S}$$

Engine dynamic model

u_{10}	u_{11}	u_{12}	u_{13}
δ_{THL}	E_{26}	δ_{THr}	E_{26}

y_{10}	y_{11}	y_{12}	y_{13}
δ_{THL1}	δ_{THL2}	δ_{THr1}	δ_{THr2}

$$G_{10\,10} = G_{12\,12} = \frac{1}{S+2.} \qquad G_{11\,11} = G_{13\,13} = \frac{1}{S+5.}$$

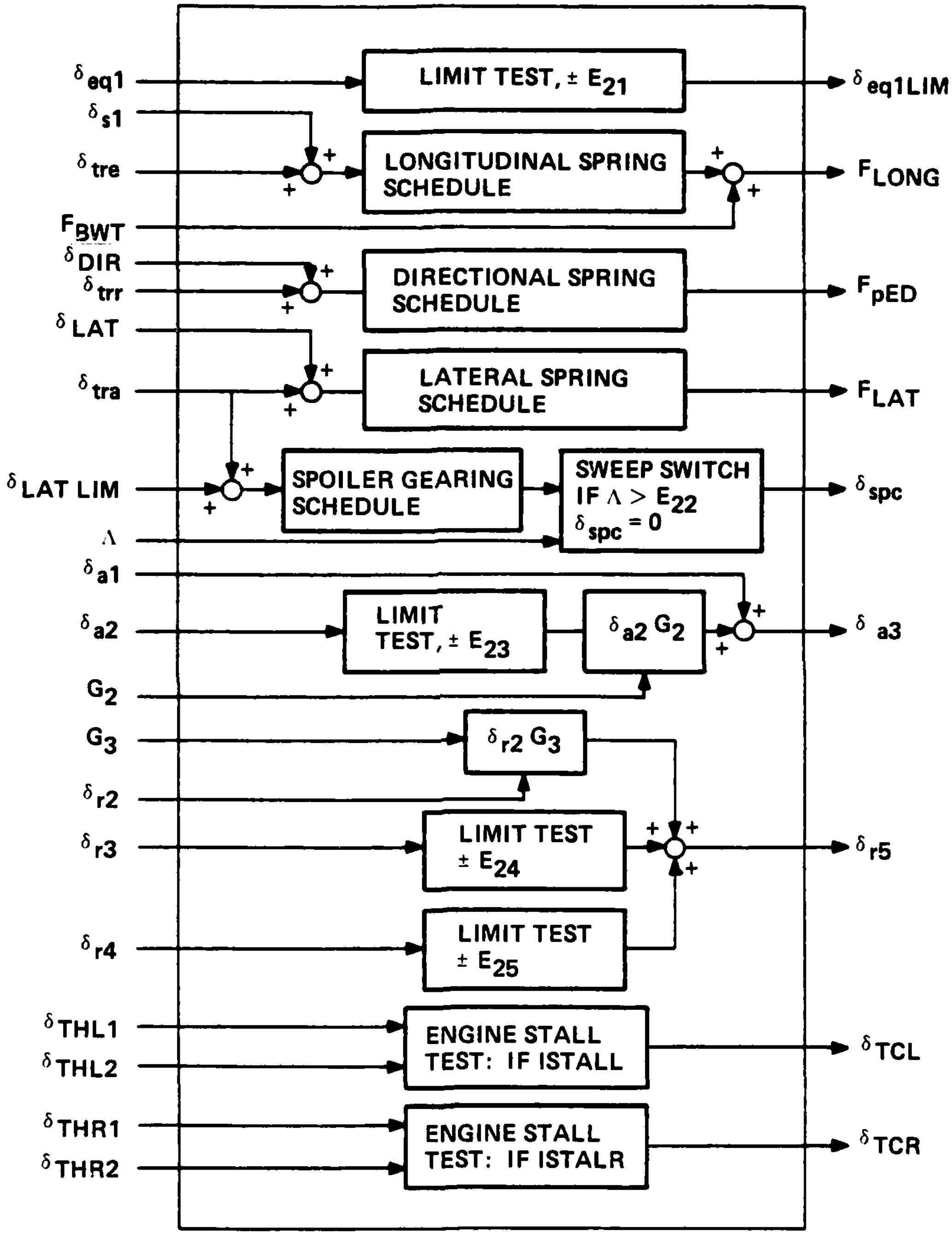


FIGURE 2. Control System A: Second Nonlinear Stage

TABLE II.

CONTROL SYSTEM A.
SECOND LINEAR STAGE

INPUTS:

u_1	u_2	u_3	u_4	u_5	u_6	u_7
$\delta eq1$ LIM	δ_{a3}	E_{32}	δspc	E_{34}	δr_5	E_{36}

OUTPUTS:

y_1	y_2	y_3	y_4	y_5	y_6	y_7
δeq_2	δa_4	δa_5	δsp	δsp_1	δr_6	δr_7

Transfer Function Matrix:

$$G_{11} = \frac{1.0114\,(S+5)^2}{(S+1.887)\,(S+13.4)}$$

$$G_{22} = G_{66} = \frac{90}{(S+90)}$$

$$G_{33} = G_{77} = \frac{90}{S}$$

$$G_{44} = \frac{20}{S+20} \qquad G_{55} = \frac{20}{S}$$

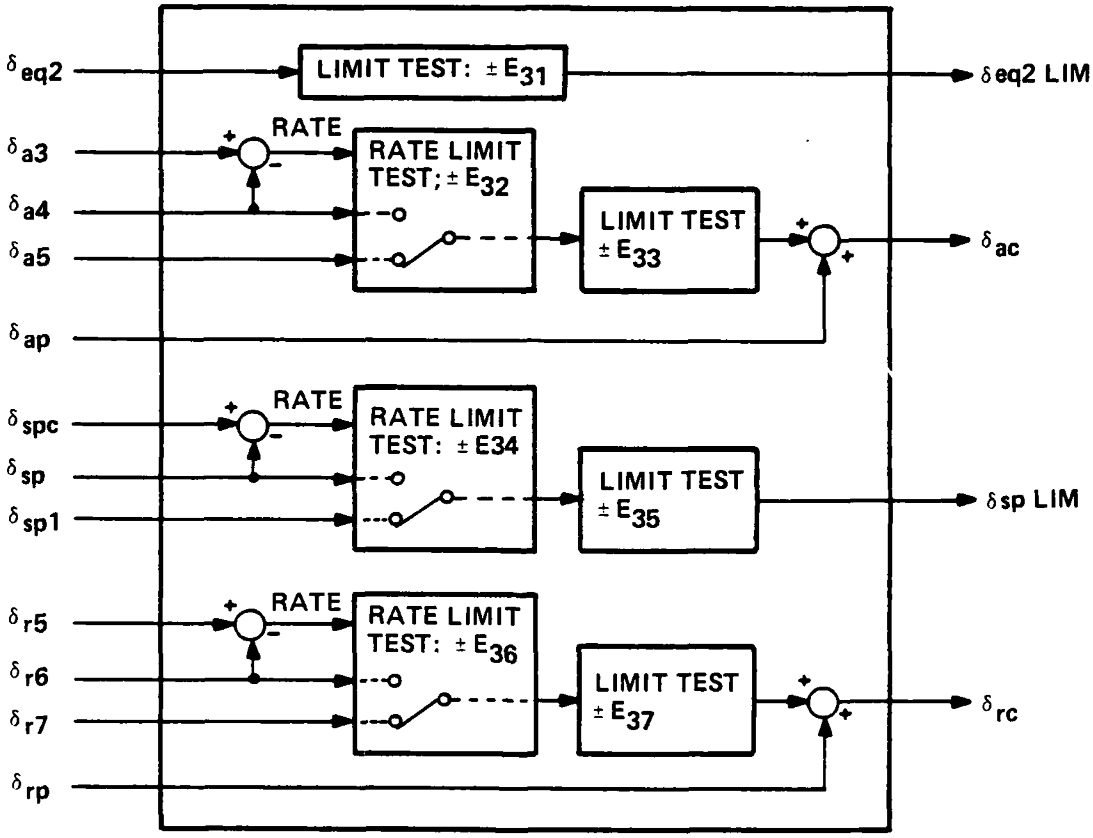


FIGURE 3. Control System A: Third Nonlinear Stage

TABLE III.

CONTROL SYSTEM A: THIRD LINEAR STAGE

INPUTS:

u_1	u_2	u_3
$\delta_{eq2\ LIM}$	δ_{rc}	E_{45}

OUTPUTS:

y_1	y_2	y_3
δ_{eq3}	δ_{r8}	δ_{r9}

Transfer Matrix Entries

$$G_{11} = \frac{66.67}{s+66.67} \quad G_{22} = \frac{2.0}{s+20} \quad G_{33} = \frac{20}{s}$$

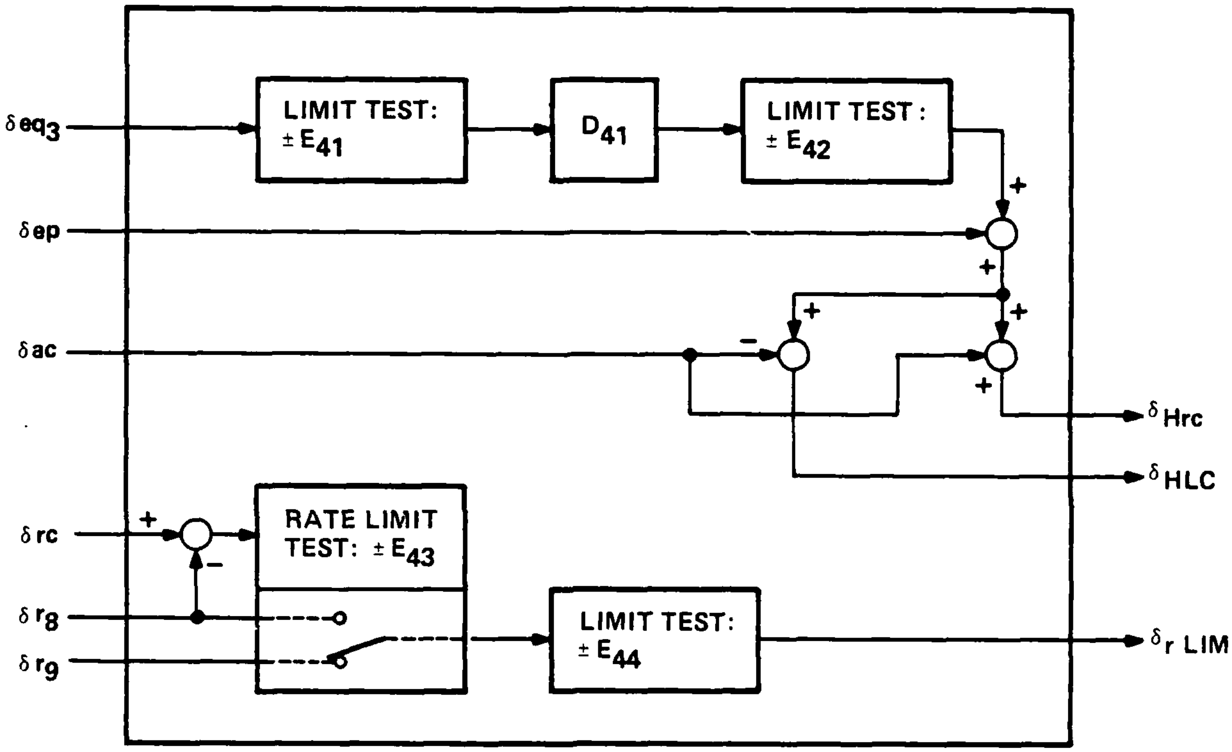


FIGURE 4. Control System A: Fourth Nonlinear Stage

TABLE IV.

CONTROL SYSTEM A: FOURTH LINEAR STAGE

INPUTS	u_1	u_2	u_3	u_4
	δ_{Hrc}	$\pm E_{51}$	δ_{HLC}	$\pm E_{51}$

OUTPUTS	y_1	y_2	y_3	y_4
	δ_{Hr1}	δ_{Hr2}	δ_{HL1}	δ_{HL2}

Transfer Matrix Entries

$$G_{11} = G_{33} = \frac{20}{s+20} \qquad G_{22} = G_{44} = \frac{20}{S}$$

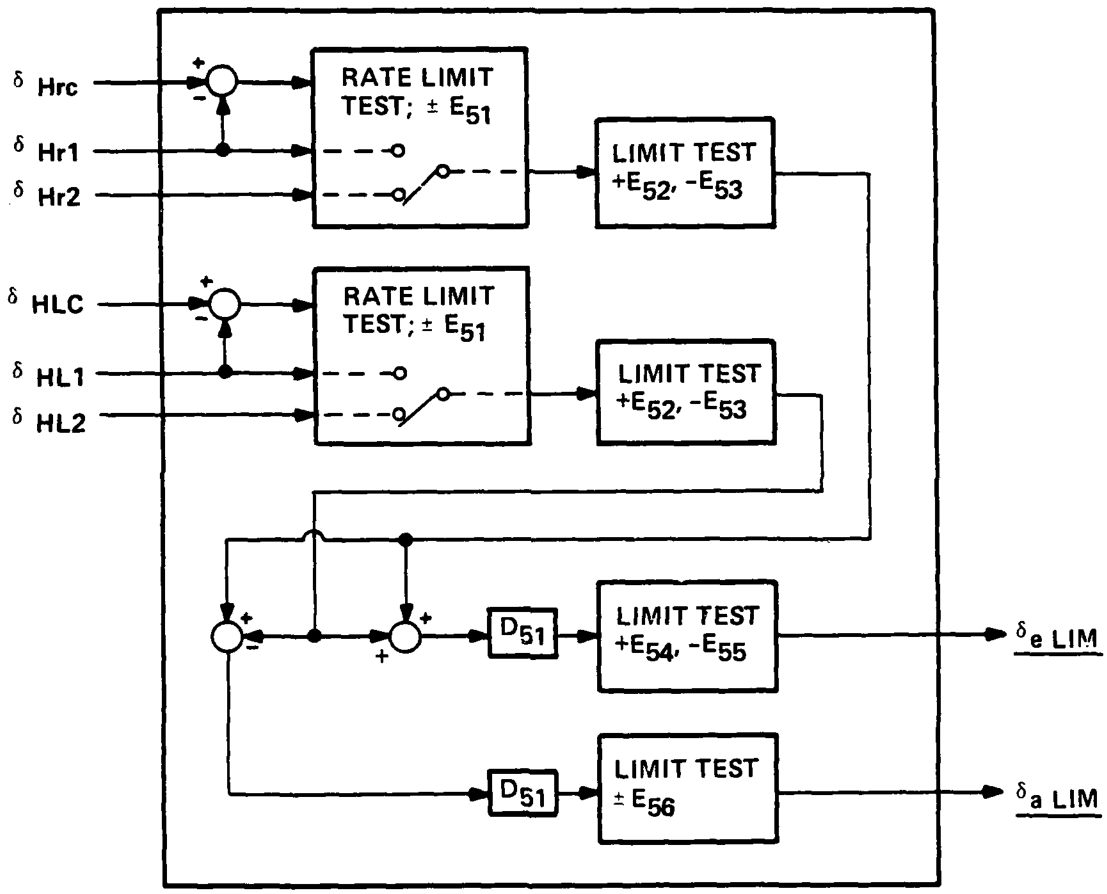


FIGURE 5. Control System A: Fifth Nonlinear Stage

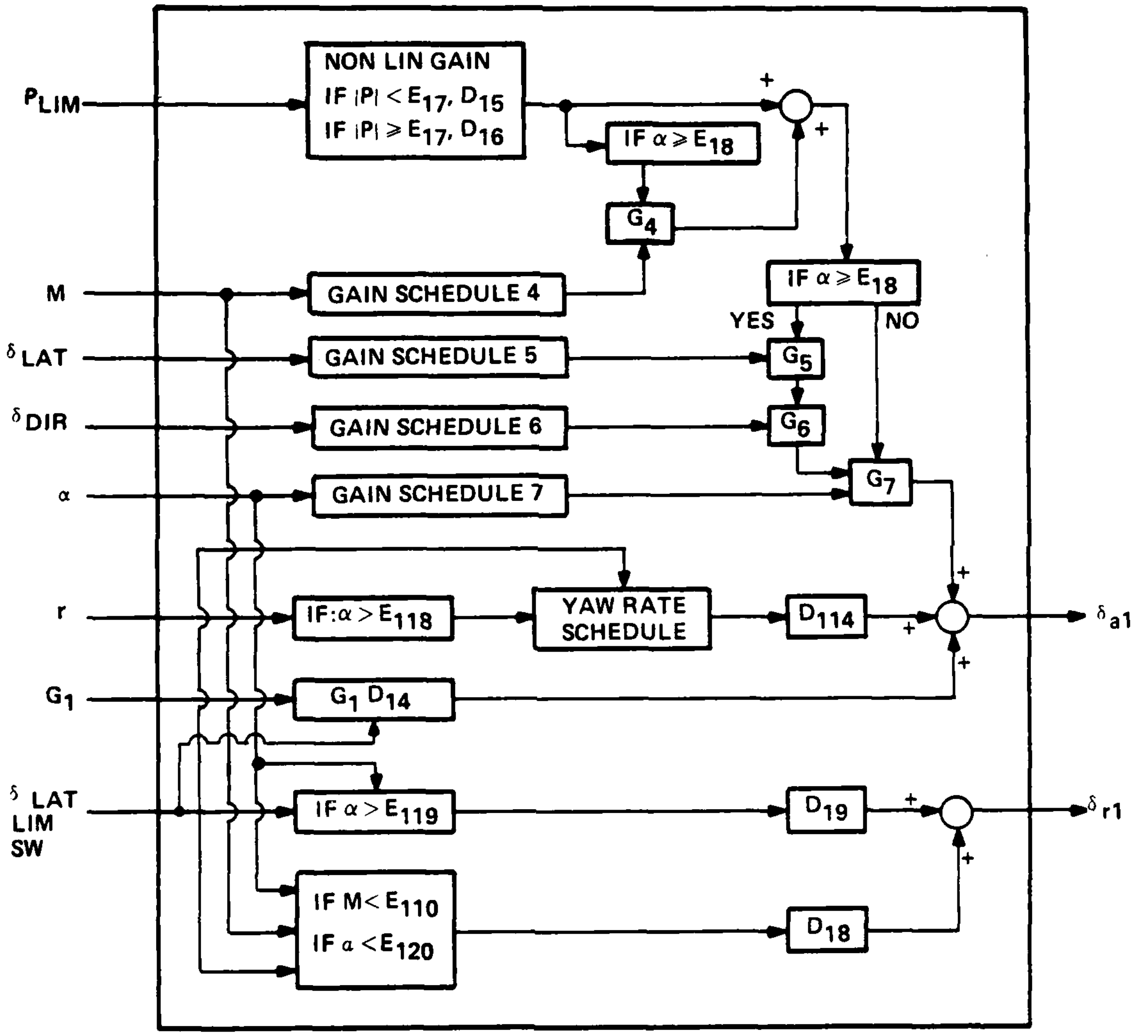


FIGURE 6. Control System B: Modifications To First Nonlinear Stage

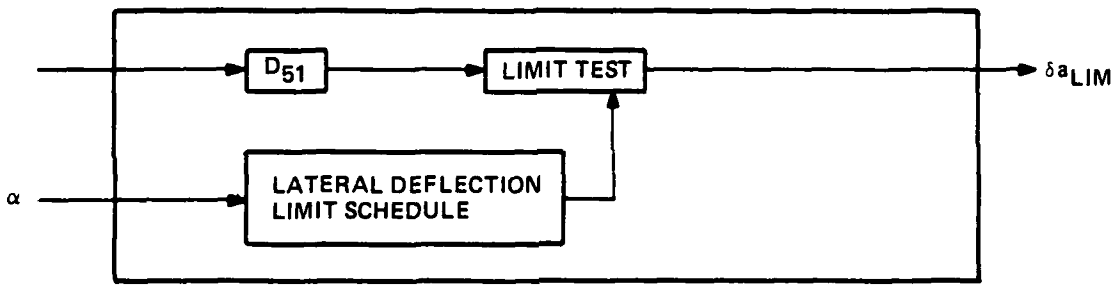


FIGURE 7. Control System B: Modifications To Fifth Nonlinear Stage

CONTROL SYSTEM CONSTANTS

TABLE Va.

CONTROL SYSTEMS A AND B

FIRST STAGE:

$E_{11} = .8727\ sec^{-1}$ (50°/sec.)	$E_{17} = 2.356\ sec^{-1}$ (135°/sec.)	$E_{1\ 13} = 1$ g.
$E_{12} = 15°$/sec.	$E_{18} = 19°$	$E_{1\ 14} = .8727\ sec^{-1}$
$E_{13} = .5$ in/sec.	$E_{19} = 1$ in.	$E_{1\ 15} = 1$ in.
$E_{14} = .19$ in/sec.	$E_{1\ 10} = .55$	$E_{1\ 16} = 1$ in.
$E_{15} = .113$ in/sec.	$E_{1\ 11} = 1$ in.	$E_{1\ 17} = .35\ sec^{-1}$
$E_{16} = 4.363\ sec^{-1}$ (250°/sec.)	$E_{1\ 12} = 1$ in.	$E_{1\ 18} = 44°$
$E_{1\ 19} = 25°$	$E_{1\ 20} = 45°$	
$D_{11} = 3$ lb./g.	$D_{16} = .15$	$D_{1\ 11} = -10°$/in.
$D_{12} = 1$ g.	$D_{17} = 2.$	$D_{1\ 12} = 9.15°$/g.
$D_{13} = 7.428$ lb-sec^2	$D_{18} = -38.96°$-sec.	$D_{1\ 13} = 22.5°$/g.
$D_{14} = 2°$/in.	$D_{19} = -12.66°$/in.	$D_{1\ 14} = -57.3°$-sec.
$D_{15} = .04$	$D_{1\ 10} = -2°$/in.	

SECOND STAGE:

$E_{21} = .2618\ sec^{-1}$ (15°/sec.)	$E_{23} = 10°$	$E_{25} = 21.5°$
$E_{22} = 55°$	$E_{24} = 50°$	

THIRD STAGE:

$E_{31} = .2618\ sec^{-1}$ (15°/sec.)	$E_{34} = 12.5°$/sec. ($\frac{250°}{20}$/sec.)
$E_{32} = .3667°$/sec. ($\frac{33°}{90}$/sec.)	$E_{35} = 55°$
$E_{33} = 5°$	$E_{36} = .7044°$/sec. ($\frac{63.4°}{90}$/sec.)
	$E_{37} = 19°$

TABLE Vb.

CONTROL SYSTEM CONSTANTS (Contd.)

FOURTH STAGE:

E_{41} = .175 sec^{-1} (10°/sec.)	E_{43} = 5.3°/sec. ($\frac{106}{20}$°/sec.)
E_{42} = 3° E_{44} = 30°	D_{41} = 17.19°-sec. (.3°/°/sec.)

FIFTH STAGE:

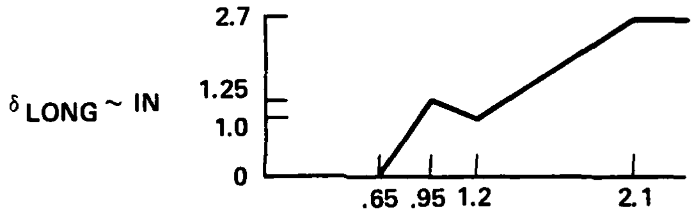
E_{51} = 1.8°/sec. ($\frac{36}{20}$°/sec.)	E_{53} = 35°	E_{55} = -33°	D_{51} = .5
E_{52} = 15°	E_{54} = 10°	E_{56} = 12°	

FOR CONTROL SYSTEM B CHANGE

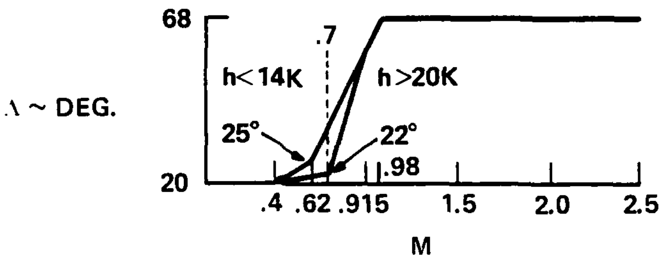
D_{19} = -6.33°/in.	D_{18} = -114.6°-sec.

CONTROL SYSTEM SCHEDULES

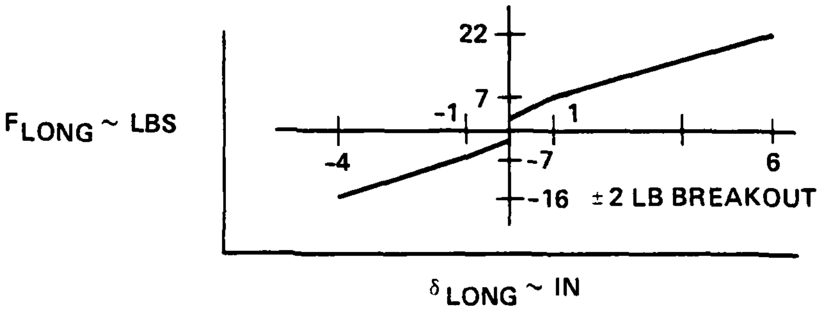
MACH TRIM:



WING SWEEP:



LONGITUDINAL SPRING:



LATERAL SPRING:

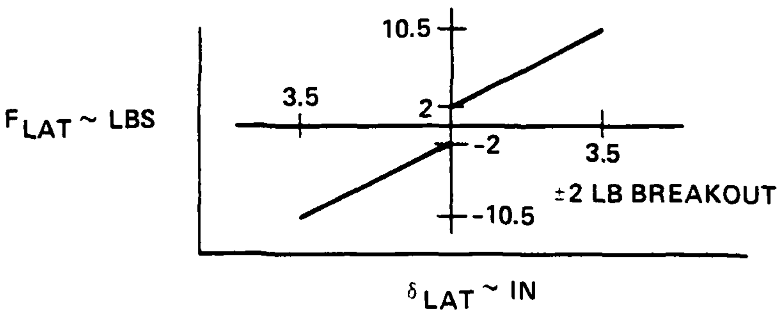
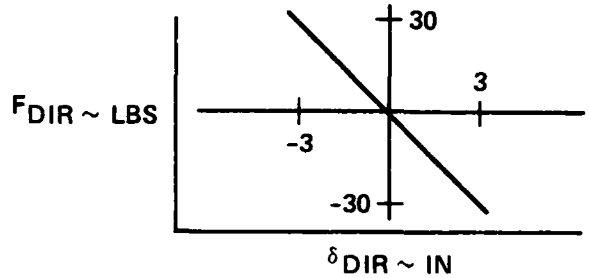
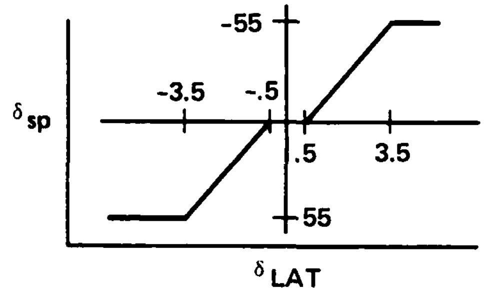


FIGURE 8a. Control Systems A & B: Schedules

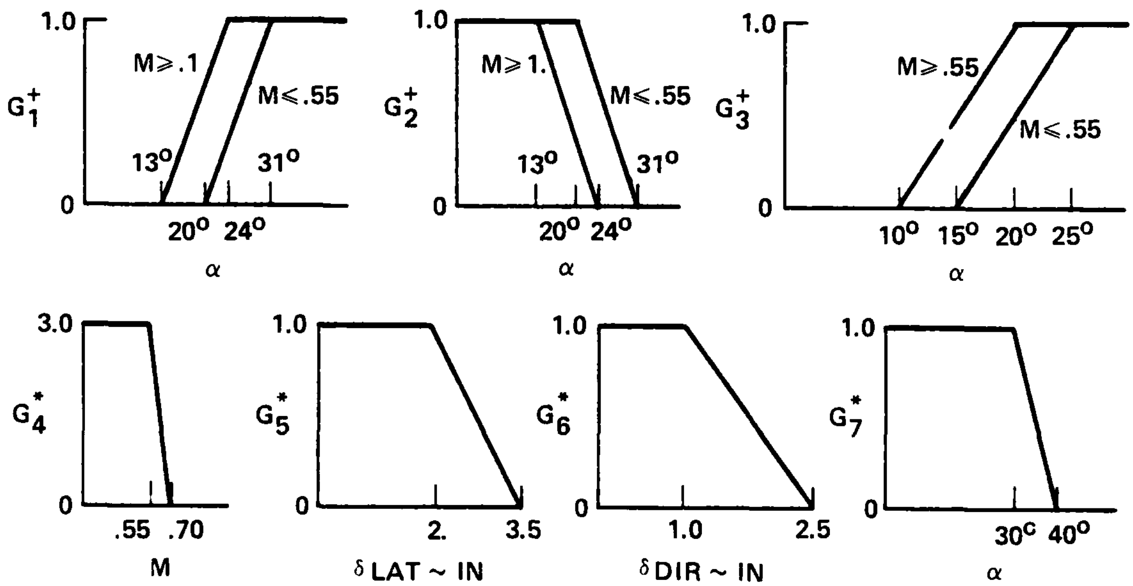
DIRECTIONAL SPRING:



SPOILER GEARING:



GAIN SCHEDULES:



+ SYSTEM A ONLY

* SYSTEM B ONLY

FIGURE 8b. Control Systems A & B, Schedules (Contd.)

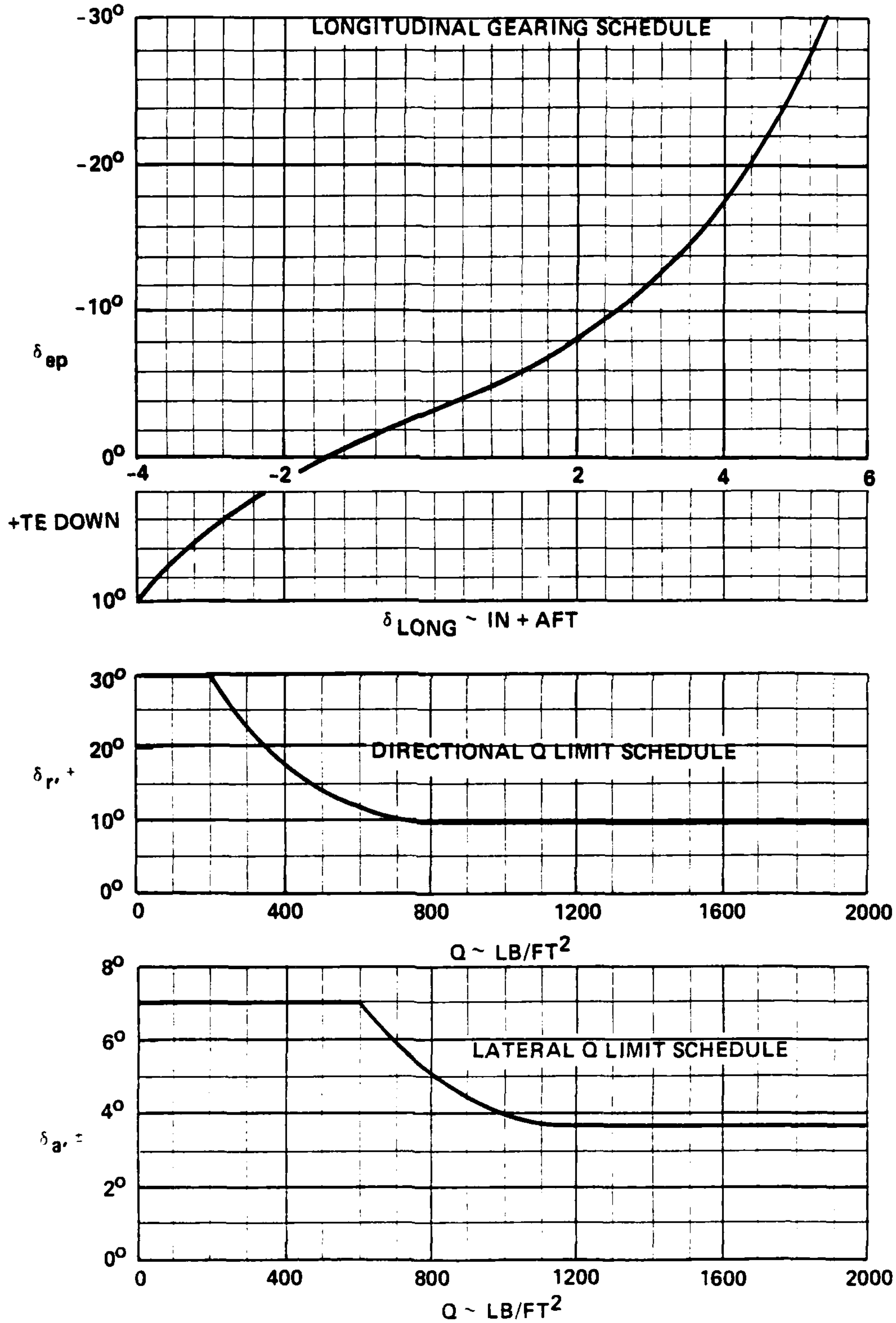


FIGURE 8c. Control Systems A & B: Schedules (Contd.)

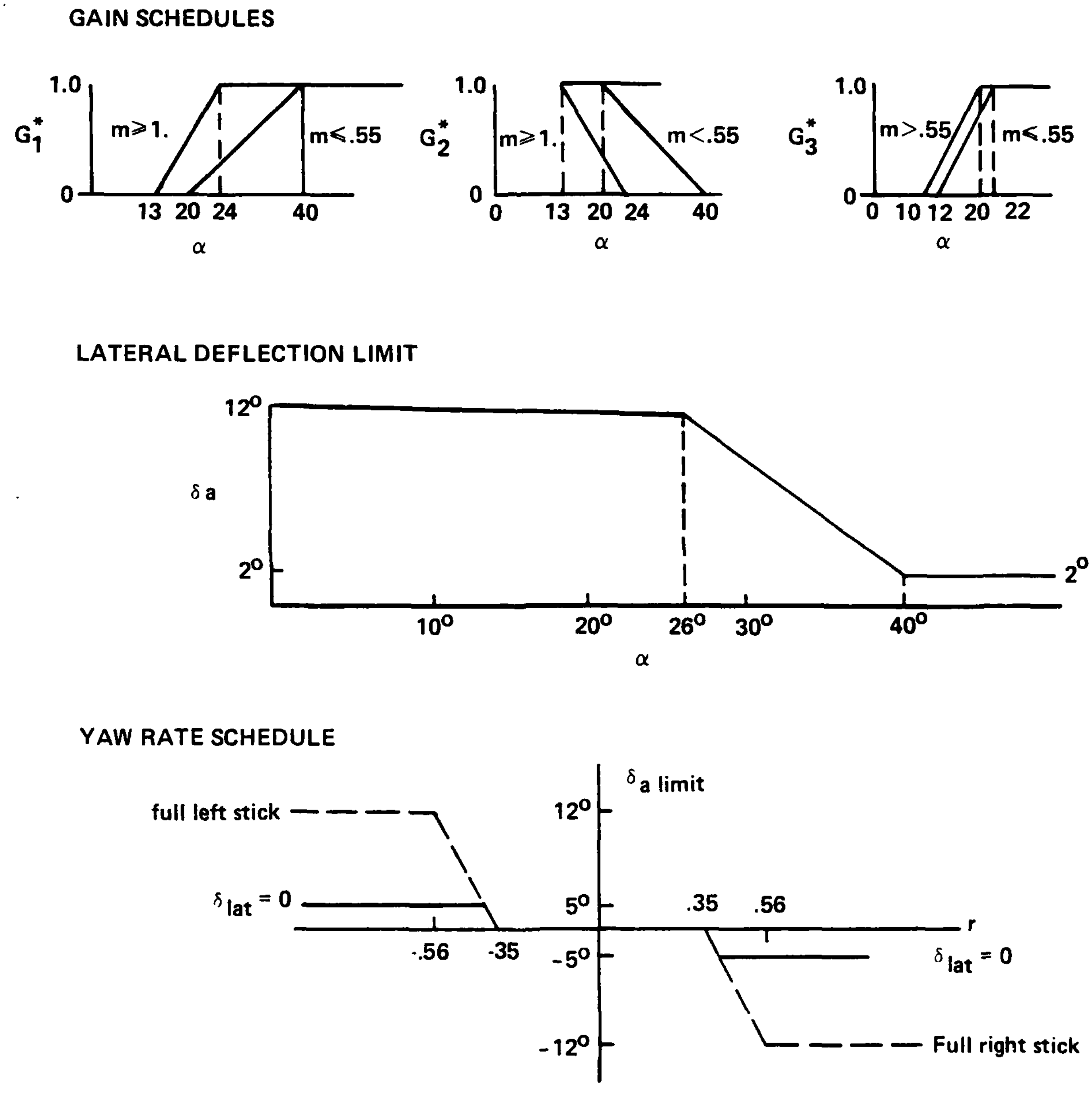


FIGURE 8d. Control Systems A & B: Schedules (Contd.)

REFERENCES

1.) NASA TM-80058; Nguyen, Ogburn, Pollock, Deal, Brown, Whipple; "Piloted Simulator Study of High-Angle-of-Attack Characteristics of F-14 Airplane with Maneuver Slats (U); May 1979.

2.) Grumman Aerospace Corp. Report No. A51-335-R-77-01 "F-14A Stability and Control and Flying Qualities Report, Status IV, Part I"; Martorella and Stiles; Dec. 16, 1977.

3.) K. Ogata; "Modern Control Engineering", Prentice Hall Inc., 1970; Ch. 13-14.

4.) AIAA Paper No. 74-884, "A Unified Approach to Digital Flight Control Algorithms". G. L. Slater; Aug. 1974.

5.) AIAA Paper; "Spin Prediction Techniques"; W. E. Bihrle & B. Barnhart; August 1980

6.) NASA Report CR-144995; "Correlation Study of Theoretical and Experimental Results of a 0.1 Scale Radio Controlled Model"; W. E. Bihrle, July 1976.

APPENDIX A

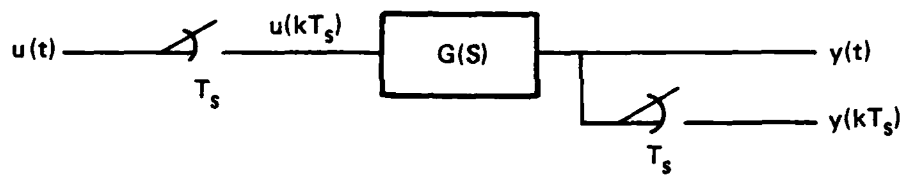
DIGITAL MODELING OF CONTINUOUS LINEAR ELEMENTS

TABLE OF CONTENTS

DIGITAL CONTROL & SIMULATION

An important difference exists between digital simulation of continuous linear elements and the design of digital control systems even though they both employ the mathematics of the discrete time domain. Digital control theory is concerned with determining the sampled or continuous response of a continuous system subjected to a sampled input.

DIGITAL CONTROL

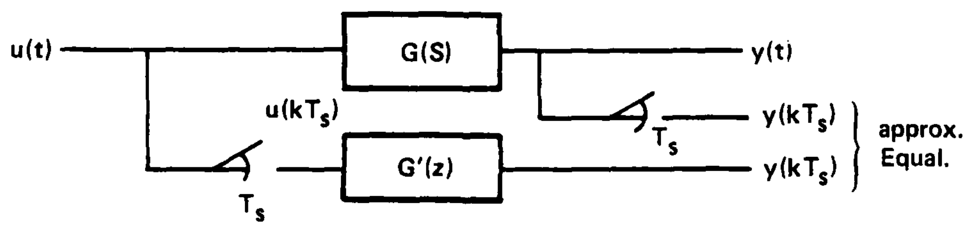


The Z transform method is directly applicable and may be used to obtain exact solutions because the actual input to the system is sampled. This result is achieved by inverse Laplace transforming the transfer function to obtain the impulse response, and then Z transforming the resultant function of time to obtain the impulse transfer function.

$$G(z) = Z\left[\mathcal{L}^{-1}\{G(s)\}\right]$$

In contrast, digital simulation seeks an impulse transfer function which when subjected to the sampled input yields a sampled output which is a good approximation to the sampled version of the continuous output signal.

DIGITAL SIMULATION

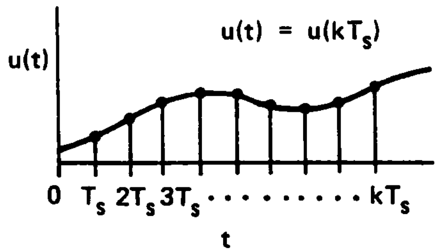


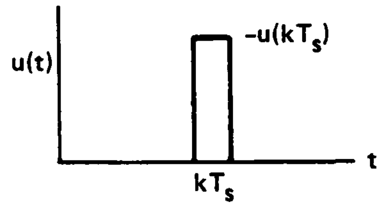
The equality can only be approximate because u(kTs) represents u(t) only at discrete instants. G'(z) must account for the variations in u(t) between sampling instants. Furthermore, since G(z) represents the response of system G(s) to sampled inputs and G'(z) the response to continuous inputs, G(z) and G'(z) are not generally equal.

For computer programming purposes, it is necessary to determine $G'(z)$ and to inverse Z transform to obtain a difference equation relating the output at any sampling instant to the previous values of input and output. To fully explain how this is done, a brief review of sampled signals and Z transforms is in order.

APPLICATION OF THE Z TRANSFORM

A sampled signal is produced by an ideal switch which opens and closes instantaneously. The switch usually closes at regular time intervals, $t=kT_S$, and remains closed for a very short period, Δt. This period is sufficiently small that the sampled signal may be considered constant during the sample period.





The sampled signal is therefore a series of square pulses of finite amplitude and very small width. The time interval Δt is imprecisely defined but constant. As such, it represents a constant scale factor which would be carried along through any linear equation without adding much to the understanding of the system. We may eliminate Δt by normalizing the square pulse, $u_s(t)$, to give

$$U^*(t) = \frac{1}{\Delta t} u_s(t).$$

Since the square pulse has the property:

$$\int_{kT_S}^{kT_S + \Delta t} u_S(t)\, dt = u(kT_S)\ \Delta t.$$

The normalized pulse has the property

$$\int_{kT_S}^{kT_S + \Delta t} u^*(t) dt = u(kT_S).$$

From this equation we suspect a close relationship between the normalized pulse and the unit impulse function, whose integral is unity, and which is comprised of two step functions whose amplitudes are adjusted as $\Delta t \rightarrow 0$.

$$\delta(t) = \mathcal{L}^{-1}\left\{\frac{1}{\Delta t}\, e^{-kT_s} \left(\frac{1 - e^{-\Delta ts}}{s}\right)\right\} \text{ as } \Delta t \to 0.$$

In order to insure u(t) constant over the interval Δt, we must allow Δt to become vanishingly small and the normalized pulse

$$u^*(t) = \mathcal{L}^{-1}\left\{\frac{u(kT_s)}{\Delta t}\, e^{-kT_s} \left(\frac{1 - e^{-\Delta ts}}{s}\right)\right\} \text{as } \Delta t \to 0.$$

Therefore, the normalized output of the sampler is a series of impulses whose strengths are equal to u(t) at each sampling instant.

$$u^*(t) = u(kT_s)\ \delta(t)$$

Laplace transforming gives

$$\mathcal{L}\left\{u^*(t)\right\} = u(kT_s)\ e^{-kT_s s}$$

If all the normalized pulses for all sampling instants are Laplace transformed and summed, and if the substitution,

$$z = e^{T_s s}$$

is made, the expression for the Z transform is obtained

$$u(z) = Z\left[u(t)\right] = \sum_{k=o}^{\infty} u(kT_s)Z^{-k}$$

The Z transform can be applied to any function of time and may be viewed as a discretized version of the Laplace transform

$$\mathcal{L}\left[u(t)\right] = \int_0^{\infty} u(t)\, e^{-st}\, dt.$$

The Z transform is a special case of the Laplace transform where the function being transformed is a series of impulses; therefore, it obeys the same algebraeic rules. It also has some unique properties of its own. One of its most important properties can be obtained by expanding the terms of its definition equation

$$u(z) = u(o) + \frac{u(T_s)}{z} + \ldots + \frac{u(nT_s)}{z^n} + \frac{u((n+1)T_s)}{z^{n+1}} + \ldots .$$

The time at which k=o, t=o is an arbitrary choice. Suppose the time where $u(t) = u(\bar{n}T_s)$ were chosen as the time for k=o, $t = t+nT_s$. The Z transform of this function expands to,

$$Z\left[u(t+nT_s)\right] = u(nT_s) + \frac{u((n+1)T_s)}{z} + \ldots\ldots$$

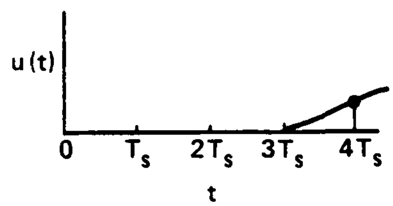
whose entries differ from the terms of u(z) for which $k \geq n$ only by the factor Z^n

$$z^n\, u(z) - Z\left[u(t+nT_s)\right] = z^n\, u(o) + z^{n-1} u(T_s) + \ldots + z\, u((n-1)T_s).$$

THE DIFFERENCE EQUATION

It will now be shown how this property allows the current value of the output to be expressed in terms of the previous values of input and output.

Recalling that the choice for t=o is arbitrary, the right hand side of the preceding equation may be made to vanish if the system was at some previous time quiescent. This may be accomplished by choosing t=o sufficiently far back in time so that the first n samples are zero. This yields the relation;



$$z^n\, u(z) = Z\left\{u(t+nT_s)\right\}.$$

More importantly, the inverse Z transform of any such function of Z is simply the value u at the n^{th} sampling instant in the future.

$$Z^{-1}\left\{z^n\, u(z)\right\} = u(t+nT_s).$$

This result is completely general and applies likewise to the output function, y(t). In general y(z) and u(z) are both infinite series but may be represented approximately by a finite number of terms. All Impulse transfer functions represent the Z transform of some system's impulse response which does not change with time. Therefore, it is always possible to obtain an approximate impulse transfer function as a ratio of constant coefficient polynomials in Z.

$$\frac{y(z)}{u(z)} = G'(Z) = \frac{P_1(z)}{P_2(z)}$$

which may be expanded to

$$P_{20}\, y(z) + P_{21}(z)y(z) + P_{22}z^2 y(z) + \cdots + P_{2m}z^m y(z) =$$

$$P_{10}\, u(z) + \cdots + P_{1n}z^n u(z)$$

which when inverse Z transformed gives

$$\sum_{i=o}^{m} P_{2i}\, y\,(t+iT_s) = \sum_{i=o}^{n} P_{1i}\; u\,(t+iT_s)$$

or alternately

$$y\,(t+mT_s) = \sum_{i=o}^{n} \frac{P_{1i}}{P_{2m}}\; u\,(t+iT_s) - \sum_{i=o}^{m-1} \frac{P_{2i}}{P_{2m}}\; y\,(t+iT_s)$$

Simply by considering the value of $y\,(t+mT_s)$ to be the current value of the output, we can make the substitution

$$t = t - mT_s$$

to obtain

$$y(t) = \sum_{i=o}^{n} \frac{P_{1i}}{P_{2m}}\; u\,(t-(m-i)T_s) - \sum_{i=o}^{m-1} \frac{P_{2i}}{P_{2m}}\; y\,(t-(m-i)T_s).$$

We have the current value of the output in terms of the previous values of input and output. However, a method for determining the approximate impulse transfer function $G'(Z)$ in terms of the ratio of constant coefficient polynomials in Z must still be developed.

CONVERSION OF TRANSFER FUNCTIONS TO DIFFERENCE EQUATIONS

A linear element is customarily defined by a transfer function that is usually expressed as the ratio of factored polynomials in S.

$$\frac{y(s)}{u(s)} = G(s) = \frac{k \prod_{i=1}^{n} (S-r_i)}{\prod_{j=1}^{m} (S-r_j)} \qquad m \geq h$$

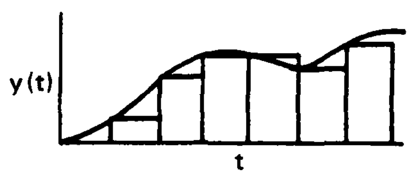
If the transfer function polynomials are expanded and divided by S^m, this equation may be stated in terms of the forms

$$\frac{a_j\, y(s)}{s^j} \quad \text{and} \quad \frac{b_i\, u(s)}{s^i} \ .$$

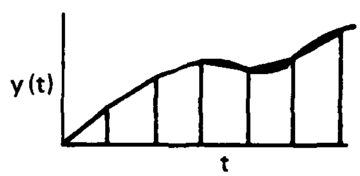
Inverse Laplace transforming these terms is equivalent to integrating a_i y(s), j times or b_i u(s), i times. If the integral can be Z transformed as a polynomial in Z times u(z) or y(z), a representation of G'(Z) of the required form will be obtained.

The sampled signal y(z) represents y(t) only at discrete instants $t = kT_S$. In order to integrate y(t) some approximation to y(t) between sampling instants must be assumed.

rectangular



trapezoidal



To obtain the value of the integral in polynomial form, y(t) must be expressed (approximately) as a polynomial.

$$y(t) \approx \sum_{i=o}^{\ell} g_i t^i .$$

Where the constants, g_i, are determined by satisfying the constraint equations

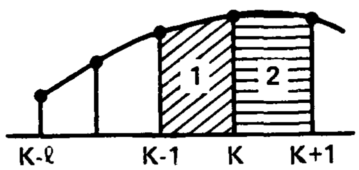
$$y(jT_s) = \sum_{i=o}^{\ell} g_i\,(jT_s)^i \qquad k \geq j \geq k - \ell$$

(k = latest sample no.)

The integral might be determined by considering the polynomial to represent y(t) over either of two intervals;

case 1: $kT_s \geq t \geq (k-1)T_s$

case 2: $(k+1)T_s \geq t \geq kT_s$



Case 1 would be expected to be more accurate for $\ell>0$, because y(t) at both ends of the interval are known. For $\ell=0$ there seems to be no particular advantage to either case.

The choice does make a difference in the final form of the Z transform. For now we will refer to the integration limits as kT_s and $(k-1)T_s$, so that

$$\int_{(k-1)T_s}^{kT_s} y(t)dt = \sum_{i=o}^{\ell} \frac{g_i}{i+1} T_s^{\,i+1} \left\{ k^{i+1} - (k-1)^{i+1} \right\}$$

over all j intervals from 1 to k

$$\int_{o}^{kT_s} y(t)\,dt = \sum_{j=1}^{k} \sum_{i=o}^{\ell} \frac{g_{ij}}{i+1} T_s^{\,i+1} \left\{ j^{i+1} - (j-1)^{i+1} \right\}$$

Z transforming, and realizing that $\int_{o}^{o} y(t)\,dt = 0$

Case 1

$$Z\left[\int_{o}^{kT_s} y(t)\,dt\right] = \sum_{k=1}^{\infty} \sum_{j=1}^{k} \sum_{i=o}^{\ell} \frac{g_{ij}}{i+1} T_s^{\,i+1} \left\{ j^{i+1} - (j-1)^{i+1} \right\} z^{-k}$$

For case 2, the limits are changed so that,

$$\int_{kT_s}^{(k+1)T_s} y(t)\,dt = \sum_{i=o}^{\ell} \frac{g_i}{i+1}\, T_s^{\,i+1} \left\{ (k+1)^{i+1} - k^{i+1} \right\} .$$

But there are now k +1 intervals

$$\int_{o}^{(k+1)T_s} y(t)\,d(t) = \sum_{j=o}^{k} \sum_{i=o}^{\ell} \frac{g_{ij}}{i+1}\, T_s^{\,i+1} \left\{ (j+1)^{i+1} - j^{i+1} \right\}$$

Also the value of the integral is not zero at k=o, since $\int_{o}^{T_s} y(t)dt \neq 0$.

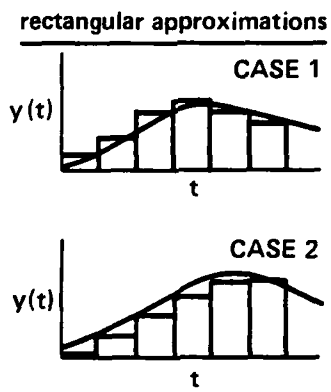
Case 2

$$Z\left[\int_{o}^{(k+1)T_s} y(t)\,dt\right] = \sum_{k=o}^{\infty} \sum_{j=o}^{k} \sum_{i=o}^{\ell} \frac{g_{ij}}{i+1}\, T_s^{\,i+1} \left\{ (j+1)^{i+1} - j^{i+1} \right\} z^{-k}$$

Although not readily apparent, this infinite series is equivalent to a geometric progression expressible as a finite polynomial in Z. Since specific expressions are needed, it is best to demonstrate this fact with an example.

Consider the simplest rectangular or zeroth order approximation to y(t).

$$\ell = 0, \; y(t) = g_{oj} = y(jT_s)$$



The Z transform of the case 1 approximation gives

$$Z\left[\int_o^{kT_s} y(t)\,dt\right] = \sum_{k=1}^{\infty} \sum_{j=1}^{k} y(jT_s)T_s\, z^{-k}$$

Whereas case 2 gives

$$Z\left[\int_o^{(k+1)T_s} y(t)dt\right] = \sum_{k=o}^{\infty} \sum_{i=o}^{k} y(jT_s)\, T_s\, z^{-k}$$

Expanding the summations

$$Z\left[\int_o^{kT_s} y(t)\,dt\right] = T_S \left\{\frac{y(T)}{z} + \frac{y(T)+y(2T)}{z^2} + \cdots\right\}$$

$$Z\left[\int_o^{(k+1)T_s} y(t)\,dt\right] = T_S \left\{y(o) + \frac{y(o)+y(T)}{z} + \cdots\right\}$$

We see the difference between the Z transform in cases 1 and 2 is

$$T_S\, y(o) \left\{1 + \frac{1}{z} + \frac{1}{z^2} + \cdots\right\}$$

The bracketed term is a geometric series which we can see has a finite expression if we allow

$$Q = 1 + \frac{1}{z} + \frac{1}{z^2} + \cdots$$

and

$$\frac{1}{z} Q = \frac{1}{z} + \frac{1}{z^2} + \frac{1}{z^3} + \cdots$$

so that, subtracting

$$Q\left(1 - \frac{1}{z}\right) = 1 \quad \text{or} \quad Q = \frac{z}{z-1},$$

Thus making the difference between cases 1 and 2

$$Z\left[\int_o^{(k+1)T_s} y(t)\,dt\right] - Z\left[\int_o^{kT_s} y(t)\,dt\right] = \frac{T_s z}{z-1}\; y(o)$$

The Z transforms themselves may be rearranged so that

$$Z\left[\int_o^{(k+1)T_s} y(t)\,dt\right] = T_S \left\{ \begin{array}{l} \left[y(o) + \frac{y(T)}{z} + \frac{y(2T)}{z^2} + \cdots\right] \\ + \frac{1}{Z}\left[y(o) + \frac{y(T)}{z} + \frac{y(2T)}{z^2} + \cdots\right] \\ + \text{ etc.} \end{array} \right\}$$

or

$$Z\left[\int_o^{(k+1)T_s} y(t)\,dt\right] = T_s y(z) \left\{1 + \frac{1}{z} + \frac{1}{z^2} + \cdots\right\}$$

Which gives for the rectangular approximation

Case 1.

$$Z\left[\int_o^{kT_s} y(t)\,dt\right] = \frac{T_s z}{(z-1)} \left\{y(z) - y(o)\right\}$$

Case 2.

$$Z\left[\int_o^{(k+1)T_s} y(t)\,dt\right] = \frac{T_s z}{(z-1)}\; y\,(z)$$

For an initially undisturbed system $y(o) = o$, and the Z transform of the integral is independent of the integration limits.

$$Z\left[\int y(t)dt\right] = \frac{T_s z}{(z-1)}\, y(z) \qquad \ell = 0.$$

Similarly for case 1 with $\ell = 1$

$$Z\left[\int_o^{kT} y(t)dt\right] = T_s \sum_{K=1}^{\infty} \sum_{j=1}^{k} \left\{ g_{oj} + g_{1j}\, T_s\, (j-1/2) \right\} z^{-k}$$

Also; $\quad y\,(jT_s) = g_{oj} + g_{1j}\,(jT_s)$

$$y\,(j-1)T_s) = g_{oj} + g_{1j}\,((j-1)T_s)$$

so that,

$$T_s g_{1j} = y(jT_s) - y\,((j-1)T_s)$$

$$g_{oj} = j\, y((j-1)T_s) - (j-1)\, y\,(jT_s)$$

Therefore

$$Z\left[\int_o^{kT_s} y(t)\, dt\right] = T_S \sum_{k=1}^{\infty} \sum_{j=1}^{k} \left\{ \begin{array}{l} \cancel{j\, y((j-1)T_s)} - j\,\cancel{y(jT_s)} + y(jT_s) \\ -j\cancel{y((j-1)T_s)} + j\cancel{y(jT_s)} - 1/2\, y(jT_s) \\ + 1/2\, y((j-1)T_S) \end{array} \right\} z^{-k}$$

$$= \frac{T_S}{2z} \sum_{k=1}^{\infty} \sum_{j=1}^{k} \left\{ y\,((j-1)T_s) + y(jT_s) \right\} z^{1-k}$$

Expanding

$$= \frac{T_S}{2z} \left\{ \begin{array}{l} y(o) + y(T_s) + \dfrac{y(o) + 2y(T_s) + y(2T_s)}{Z} \\ + \dfrac{y(o) + 2y(T_s) + 2y(2T_s) + y(3T_s)}{Z^2} + \ldots \end{array} \right\}$$

Separating out:

$$y(Z) = y(o) + \frac{y(T_s)}{Z} + \frac{y(2T_s)}{Z^2} + \cdots$$

reduces to

$$= \frac{T_S}{2z} \left\{ \begin{array}{l} y(z) \ (1 + 1/z + 1/z^2 + \cdots) \\ + \ y(T_s) \ (1 + 1/z + 1/z^2 + \cdots) \\ + \ \frac{y(2T_s)}{z} \ (1 + 1/z + 1/z^2 + \cdots) \\ + \ \ldots \text{ etc.} \end{array} \right\}$$

$$= \frac{T_S}{2(z-1)} \left\{ y(z) + Z \left[y(t+T_s) \right] \right\}.$$

From the property of Z transforms on page A-4, with n=1 we can write

$$Z\left[\int_o^{kT_s} y(t)\, dt\right] = \frac{T_S}{2\,(z-1)} \left\{ (z+1)\, y(z) - zy(o) \right\}$$

Setting $y(o) = 0$ and removing the integration limits:

$$Z\left[\int y(t)\, dt\right] = \frac{T_S\,(z+1)}{2\,(z-1)}\, y(z) \qquad \ell = 1$$

With considerably more labor we can obtain

$$Z\left[\int y(t)\, dt\right] = \frac{T_S}{3}\, \frac{(z^2-1)\, y(z)}{(z^2 + 4z+1)} \qquad \ell = 2.$$

We might generalize that for the ℓ^{th} order approximation to $y(t)$

$$Z\left[\int y(t)\, dt\right] = \frac{T_S}{\ell+1}\, \frac{q_2(z)}{q_1(z)}\, y(z)$$

where $q_1(z)$ and $q_2(z)$ are ℓ^{th} order polynomials in z for $\ell > 0$.

In going from the Laplace domain to the Z domain, we may write;

$$Z\left[\mathcal{L}^{-1}\left\{\frac{a_i}{s}\,y(s)\right\}\right] = Z\left[a_i\int y(t)\,dt\right] = \frac{a_i T_s}{\ell+1}\;\frac{q_2(z)}{q_1(z)}\;y(z)$$

which is equivalent to simply changing y(s) to y(z) and substituting.

$$s = \frac{\ell+1}{T_s}\;\frac{q_1(z)}{q_2(z)}$$

The principle is extendable to multiple integration so that

$$s^i = \frac{(\ell+1)^i}{T_s^{\,i}}\;\frac{q_1^{\,i}(z)}{q_2^{\,i}(z)}$$

and the substitution algorithm has general applicability.

$$G'(z) = G(s)\Big|_{\,s = \frac{\ell+1}{T_S}\frac{q_1(z)}{q_2(z)}}$$

Since G(s) is the ratio of finite polynomials in s, this substitution produces a ratio of finite polynomials in z which when inverse Z transformed express y(t) as on page A-5 as

$$y(t) = \sum_{i=o}^{n} \frac{P_{1i}}{P_{2m}}\,u(t-(m-i)T_s) - \sum_{i=o}^{m-1} \frac{P_{2i}}{P_{2m}}\,y(t-(m-1)T_s).$$

USE OF POLYNOMIAL EXPANSION AND THE TUSTIN TRANSFORMATION

To relate the coefficients of this equation to the transfer function roots and to the order of the approximation to y(t) and u(t), we need only have a general method of expanding polynomials. Such a method can be found by expressing a polynomial after ℓ factors have been multiplied out as;

$$P_\ell(s) = \sum_{i=o}^{\ell} a_{\ell i}\,s^i \quad \text{where} \quad a_{00} \doteq 1.$$

All additional factors may be broken into first or second order factors. If the next term is first order,

$$\left.\begin{aligned} P_{\ell+1}(s) &= (s + 1/\tau)\ P_\ell\ (s) = \sum_{i=0}^{\ell+1} a_{(\ell+1)}^{\ i}\ s^i \\ P_{\ell+1}(s) &= \sum_{i=o}^{\ell+1} \left(a_{\ell(i-1)} + \frac{a_{\ell i}}{\tau}\right) s^i \end{aligned}\right\} \begin{aligned} & a'_{\ell i} = 0 \\ & \text{if } i > \ell \\ & \text{or } i < 0 \end{aligned}$$

If the next term is second order

$$P_{\ell+2}(s) = (s^2 + 2\zeta\omega_n s + \omega_n^2)\ P_\ell\ (s)$$

$$P_{\ell+2}(s) = \sum_{i=o}^{\ell+2} (a_{\ell i}\,\omega_n^2 + 2\zeta\omega_n\ a_{\ell\ (i-1)} + a_{\ell(i-2)})\ s^i$$

If we make the substitution

$$s = \frac{\ell+1}{T_S}\ \frac{q_1(z)}{q_2(z)}$$

into the n^{th} order polynomial is S, we obtain an ℓn^{th} order polynomial in Z, and we must keep track of the last ℓn values of input and output to calculate y(t). There is therefore a tradeoff between complexity and accuracy in selecting a value for ℓ. The substitution for ℓ=1 has been used extensively as a good compromise value and is usually called the Tustin transformation.*

$$s = \frac{2}{T_S}\ \frac{(z-1)}{(z+1)}$$

If a transfer functions n^{th} order numerator and m^{th} order denominator are expanded into polynomials of s, the Tustin transform may be applied to give

*See reference 4

$$y(z)\sum_{i=o}^{m} a_i \left[\frac{2\,(z-1)}{T_S\,(z+1)}\right]^i = u(z)k \sum_{i=o}^{n} b_i \left[\frac{2\,(z-1)}{T_S\,(z+1)}\right]^i$$

where k is the transfer function gain.

Multiplying both sides of this equation by

$$T_s^{\,m}\ (z+1)^m$$

gives:

$$y(z)\sum_{i=o}^{m} a_i\, 2^i\, T_s^{\,m-i}\, (z-1)^i\, (z+1)^{m-i} =$$

$$kU(z)\sum_{i=o}^{n} b_i\, 2^i\, T_s^{\,m-i}\, (z-1)^i\, (z+1)^{m-i}.$$

The result is that both sides of the equation contain m^{th} order polynomials in z. If the expansion of the i^{th} set of z factors is defined as

$$\sum_{j=o}^{m} h_{ij}\, z^j = (z-1)^i\, (z+1)^{m-i}$$

h_{ij} for all values of $m \geq i \geq o$ may be calculated using the factor expansion process developed on page A-14.

This then makes the expanded version of the impulse transfer function (G'(z))

$$y(z)\sum_{i=o}^{m}\sum_{j=o}^{m} a_i\, 2^i\, T_s^{\,m-i}\, h_{ij}\, z^j = ku(z)\sum_{i=o}^{n}\sum_{j=o}^{m} b_i\, 2^i\, T_s^{\,m-i}\, h_{ij}\, z^j$$

Making the coefficients of the difference equation

$$y(t) = \sum_{j=o}^{m} k \frac{P_{1j}}{P_{2m}} u(t-(m-j)T_s) - \sum_{j=o}^{m-1} \frac{P_{2j}}{P_{2m}} y(t-(m-j)T_s)$$

equal to

$$P_{1j} = \sum_{i=o}^{n} b_{ni} 2^i T_s^{m-i} h_{ij}$$

$$P_{2j} = \sum_{i=o}^{m} a_{mi} 2^i T_s^{m-i} h_{ij}$$

Thus the difference equation representing the inverse Z transform of G'(z) may be obtained directly from the transfer function G(s).

In order to calculate y(t) directly using this formulation, we require the storage of 2m-1 coefficients and a like number of past input and output values for each element. Significant efficiencies in representing single and multiple element systems can be achieved using the state space notation of appendix B. This method minimizes the variables and algebraeic manipulations required to calculate all system outputs while also presenting a very flexible means of representing the system element structure.

In order to demonstrate this formulation two simple examples are now presented.

Example 1: First order filter

$$G(s) = \frac{k}{(S+\sigma)} \quad n=0, \ m=1$$

Using the expansion procedure

$$a_{00} = 1 \qquad b_{00} = 1$$

$$P_1(s) = \sum_{i=o}^{1} (a_{o\,i-1} + a_{oi}\ \sigma)\, s^i = \sum_{i=o}^{1} a_{1i}\ s^i$$

$$a_{10} = \cancel{a_{0-1}} + a_{00}\ \sigma = \sigma$$

$$a_{11} = a_{00} + \cancel{a_{01}}\ \sigma = 1$$

Also expanding

$$(Z-1)^i (Z+1)^{1-i} = \sum_{j=o}^{m} h_{ij} Z^j, \ 1 \geq i \geq o$$

gives $h_{00} = 1$ $\qquad h_{01} = 1$

$h_{10} = -1$ $\qquad h_{11} = 1$

Evaluating the difference equation coefficients

$$P_{10} = b_{00}\, T_S\, h_{00} = T_S \qquad P_{11} = b_{00}\, T_S\, h_{01} = T_S$$

$$P_{20} = a_{10}\, T_S\, h_{00} + a_{11}\, 2\, h_{10} = (\sigma T_S - 2)$$

$$P_{21} = a_{10}\, T_S\, h_{01} + a_{11}\, 2\, h_{11} = (\sigma T_S + 2)$$

making the difference equation.

$$y(t) = \frac{k}{(\sigma + 2/T_S)} \left\{ u(t) + u(t-T_S) \right\} - \frac{(\sigma T_S - 2)}{(\sigma T_S + 2)}\ y\,(t-T_S)$$

<u>Example 2: First over second order system</u>

$$G(s) = \frac{k\ (s+\sigma)}{(s^2 + \underline{2\zeta s} + \omega_n^2)}_{\ \omega_n} \qquad n=1,\ m=2$$

Expanding the numerator polynomial as with example 1's denominator

$$b_{10} = \sigma, \ b_{11} = 1.$$

Expanding the denominator

$$a_{00} = 1$$

$$P_2(s) = \sum_{i=o}^{2} \left\{ a_{oi}\,\omega_n^{\;2} + 2\zeta\omega_n\, a_{o\,i-1} + a_{o\,i-2} \right\} s^i$$

$$a_{20} = a_{00}\,\omega_n^{\;2} + 2\zeta\omega_n \cancel{a_{0-1}} + \cancel{a_{0-2}} = \omega_n^{\;2}$$

$$a_{21} = \cancel{a_{01}}\,\omega_n^{\;2} + 2\zeta\omega_n\, a_{00} + \cancel{a_{0-1}} = 2\zeta\omega_n$$

$$a_{22} = \cancel{a_{02}}\,\omega_n^{\;2} + \cancel{2\zeta\omega_n a_{01}} + a_{00} = 1$$

Expanding

$$(z-1)^i\,(z+1)^{m-i} = \sum_{j=0}^{m} h_{ij}\, z^j \qquad 2 \geq i \geq 0$$

for i = o

$$(Z+1)\,(Z+1) = \sum_{j=0}^{2} h_{oj}\, Z^j$$

using the notation

$$\ell h_{ij} = h_{ij} \text{ after } \ell \text{ factors expanded}$$

$$0h_{00} = 1$$

$$1h_{00} = 0\,\cancel{h_{0-1}} + (0h_{00} \times 1) = 1$$

$$1h_{01} = 0h_{00} + (\cancel{0h_{01}} \times 1) = 1$$

$$h_{00} = 2\,h_{00} = 1\cancel{h_{0-1}} + (1h_{00} \times 1) = 1$$

$$h_{01} = 2\,h_{01} = 1h_{00} + (1h_{01} \times 1) = 2$$

$$h_{02} = 2\,h_{02} = 1h_{01} + (1\cancel{h_{02}} \times 1) = 1$$

which can be verified by

$$(z+1)^2 = z^2 + 2z + 1$$

Without bothering to demonstrate the expansion formula for i=1 and i=2 we have

$$h_{10} = -1 \qquad h_{11} = 0 \qquad h_{12} = 1$$

$$h_{20} = 1 \qquad h_{21} = -2 \qquad h_{22} = 1$$

The difference equation coefficients are:

$$P_{10} = b_{10}\,T_S^2\,h_{00} + b_{11}\,2\,T_S\,h_{10}$$

$$= \sigma T_S^2 - 2\,T_S = T_S\,(\sigma T_S - 2)$$

$$P_{11} = b_{10}\,T_S^2\,h_{01} + b_{11}\,2\,T_S\,h_{11}$$

$$= 2\sigma T_S^2$$

$$P_{12} = b_{10}\,T_S^2\,h_{02} + b_{11}\,2\,T_S\,h_{12}$$

$$= \sigma T_S^2 + 2\,T_S = T_S\,(\sigma T_S + 2)$$

$$P_{20} = a_{20}\,T_S^2\,h_{00} + a_{21}\,2\,T_S\,h_{10} + a_{22}\,4\,h_{20}$$

$$= \omega_n^2\,T_S^2 - 4\,\zeta\,T_S/\omega_n + 4$$

$$P_{21} = a_{20}\,T_S^{\,2}\,h_{01} + a_{21}\,2\,T_S\,h_{11} + 4\,a_{22}\,h_{21}$$

$$= 2\,(\omega_n^{\,2}\,T^2 - 4)$$

$$P_{22} = a_{20}\,T_S^{\,2}\,h_{02} + a_{21}\,2\,T_S\,h_{12} + a_{22}\,4\,h_{22}$$

$$= \omega_n^{\,2}\,T_S^{\,2} + \frac{4\zeta\,T_s}{\omega_n} + 4.$$

Making the difference equation

$$y(t) = \left[\frac{k\,T_S\,(\sigma T_S - 2)}{\omega_n^{\,2}\,T_S^{\,2} + \frac{4\,\zeta\,T_S}{\omega_n} + 4}\right] u\,(t-2T_S)$$

$$+ \left[\frac{2k\,\sigma T_S^{\,2}}{\omega_n^{\,2}\,T_S^{\,2} + \frac{4\,\zeta\,T_S}{\omega_n} + 4}\right] u\,(t-T_S)$$

$$+ \left[\frac{k\,T_S\,(\sigma T_S + 2)}{\omega_n^{\,2}\,T_S^{\,2} + \frac{4\,\zeta\,T_S}{\omega_n} + 4}\right] u(t)$$

$$- \left[\frac{\omega_n^{\,2}\,T_S^{\,2} - 4\,\zeta\,T_S/\omega_n + 4}{\omega_n^{\,2}\,T_S^{\,2} + 4\,\zeta\,T_S/\omega_n + 4}\right] y\,(t-2T_S)$$

$$- \left[\frac{2\,(\omega_n^{\,2}\,T^2 - 4)}{\omega_n^{\,2}\,T_S^{\,2} + 4\,\zeta\,T_S/\omega_n + 4}\right] y\,(t-T_S)$$

APPENDIX B

APPLICATION OF STATE SPACE NOTATION

Table of Contents

THE STATE SPACE CONCEPT

Any time invariant linear system may be described in the continuous time domain by a linear ordinary differential equation of the form.

$$\frac{d^m y}{dt} + a_1 \frac{d^{m-1} y}{dt^{m-1}} + \cdots + a_{m-1} \frac{dy}{dt} + a_m y = b_o \frac{d^n u}{dt^n} + \cdots + b_n u \qquad n \geq m$$

or in the discrete time domain by a difference equation of the form.

$$y(t) + \frac{P_{2\,m-1}}{P_{2m}} y\,(t - T_s) + \cdots + \frac{P_{20}}{P_{2m}} y\,(t - mTs) = \frac{kP_{1m}}{P_{2m}} u(t) + \cdots + \frac{kP_{10}}{P_{2m}} u\,(t - mTs)$$

The similarity of these forms suggests that either equation may be considered a linear combination of 2m dependent variables. Those in the continuous domain are related by differentiation, those in the discrete domain by a time shift. The analogy is complete in the Laplace and Z domains since

$$\mathcal{L}\left[a_{m-1} \frac{d^i y}{dt}\right] = a_{m-1} s^i y(s) \text{ and } z\left[\frac{P_{2i}}{P_{2m}} y\,(t - (m-i)\,T_s)\right] = \frac{P_{2i}}{P_{2m}} z^{i-m} y\,(z)$$

Thus realizing that we may interchange

$$s^i \Leftrightarrow z^{i-m}, \quad a_i \Leftrightarrow \frac{P_{2\,m-i}}{P_{2m}}, \quad b_i \Leftrightarrow \frac{kP_{1\,m-i}}{P_{2m}}, \text{ etc.}$$

we may adopt a notation convention for these two equations such that for any function g;

$$g^j = \frac{d^j g}{dt^j} \quad \text{or} \quad g^j = g\,(t - (m-j)\,T_s). \qquad j = m-i$$

The differential and difference equations now have a common form which may be expressed in either of two useful forms. The coefficients of the differential equations are used for simplicity.

$$(a_m y - b_m u) + (a_{m-1} y' - b_{m-1} u') + \cdots + (y^m - b_o u^m) = 0 \qquad \text{Eq. .1}$$

or

$$\underbrace{y}_{\text{Output}} = \underbrace{\frac{1}{a_m}} \sum_{i=1}^{m} \underbrace{b_{m-i} u^i - a_{m-i} y^i}_{\text{State}} + \underbrace{\frac{b_m}{a_m} u}_{\text{Input}} \qquad \text{Eq. .2}$$

The output, y, is dependent only on the input, u, and the first m higher orders of y and u. If we are only interested in the output value, we may set all orders higher than m^{th} to any value, especially zero, without affecting the output calculation. In doing so, we cause equations 1 and 2 to imply a whole system of m+1 equations obtained by differentiating or shifting equation 1 m times.

<u>Equation Set A</u>

$$(a_m y - b_m u) + (a_{m-1} y^1 - b_{m-1} u^1) + \cdots + (y^m - b_o u^m) = 0$$

$$(a_m y^1 - b_m u^1) + (a_{m-1} y^2 - b_{m-1} u^2) + \cdots + (a_1 y^m - b_1 u^m) = 0$$

$$\vdots$$

$$(a_m y^{m-1} - b_m u^{m-1}) + (a_{m-1} y^m - b_{m-1} u^m) = 0$$

$$a_m y^m - b_m u^m = 0$$

In the case where set A represents difference equations, it is clear that the various orders of y and u represent the past history of the input and output. The same is true for the differential equation interpretation of set A since the derivatives of y and u represent those time histories by virtue of their Taylor's series expansions.

Because an initially quiescent system requires some type of input to disturb the system and produce an output, output may intuitively be considered a direct consequence of input. But the system equation indicates the output also depends on the past history of both input and output, by virtue of the higher order terms in y and u.

This intuitive notion leads to the concept of system state as a combination of all factors, distinct from the input and associated with past events, which contribute to the output. The state may further be defined by a set of state variables whose values, along with that of the input, determine the output. It is clear from equation 2 that state variables, x_i, may be defined so that

$$y = \Sigma\, C_i x_i + Du$$

which is most often written in matrix notation as

$$y = C\bar{x} + Du$$

where C is a row vector, D is a constant, and $\bar{x}$ a column vector (the state vector) whose components are the state variables. What is neither clear, nor in fact unique, is how to associate the state variables with the terms of equation 1.

A logical first step in establishing such an association is to determine how many state variables are required. Equation set A represents all the equations necessary to describe the system. Since there are m+1 equations, m+1 variables are required to make the set determinate. That is, set A may imply the value of an m+2nd variable if it can be restated in terms of m+1 other variables whose values are known. Since y is the variable whose value is sought, it may be considered the m+2nd variable whose value, according to the state space form of equation 2, is determinable directly from u and the state variables. Therefore, there must be a minimum of m state variables, and any additional state variables are unnecessary and may be eliminated.

Consequently, if they are properly defined, there will be m state variables which are linearly independent since none may be eliminated. Therefore, the state space notation is more efficient than a straightforward application of the difference equation because it reduces the number of variables (and their associated manipulations) from 2(m+1) to m+2.

The state space form of equation 1,

$$y = C\bar{x} + Du$$

may be considered a partial definition of the state variables which specifies the relationship

$$C\bar{x} = \sum_{i=1}^{m} \frac{b_{m-i}}{a_m} u^i - \frac{a_{m-i}}{a_m} y^i + \left(\frac{b_m}{a_m} - D \right) u$$

But one such equation is not enough to define the constant D and the m state variables which do not appear in equation set A. An additional m equations are required which must not only insure the independence of the state variables but must be consistent with set A in order to represent the systems. That is, $\bar{x}$ and D may be defined by any set of m+1 equations that are linearly independent and related to each of the variables of set A by some linear combination.

This principle allows many possible formulations of the additional definition equations. Highly arbitrary choices may be involved in the selection of any particular form. Simplicity then becomes the guiding principle in such choices.

If we retain the form of

$$y = C\bar{x} + Du$$

for the additional m equations we have

$$f_1 = A_1\bar{x} + B_1 u$$

$$f_2 = A_2\bar{x} + B_2 u$$

$$\vdots$$

$$f_m = A_m\bar{x} + B_m u$$

or in matrix form.

$$\bar{F} = A\,\bar{x} + Bu$$

The independence of the state variables may be assured by requiring

$$\det A \neq 0$$

This follows from the property of determinants which allows any constant multiple of one row to be added to another without changing the value of the determinant. If any state variable is dependent, it may be expressed as some linear combination of other state variables. It then would be possible to manipulate the rows of A in this manner until either one row or one column is all zeroes. Again by the properties of determinants, det A = 0 under these conditions.

To insure the definition equations are consistent with set A, we require each variable of set A be expressible as a linear combination of the variables requiring definition (i.e., x_i and D). If $\bar{v}$ represents the variables of set A, $\bar{w}$ the variables of the definition set, and Q the matrix relating the two.

$$\bar{v} = Q\bar{w}$$

Furthermore, if $P\bar{v}$ represents equation set A, each equation of set A may be directly equated with a linear combination of x_i and D.

$$P\bar{v} = PQ\bar{w} = R\bar{w}$$

A very straightforward way of insuring consistency is to impose a direct correspondence between the equations of set A, and the lines of

$$\left.\begin{aligned} y &= C\bar{x} + Du \\ \bar{F} &= A\bar{x} + Bu \end{aligned}\right\} \text{definition set.}$$

In equation set A, each line is obtained by incrementing the equation order and dropping out terms higher than m^{th} order. The same must be true of the definition set.

<u>Equation Set B</u>

$$-y^1 + C\bar{x}^1 + Du^1 = -f_1 + A_1\bar{x} + B_1u$$

$$-f_1^1 + A_1\bar{x}^1 + B_1u^1 = -f_2 + A_2\bar{x} + B_2u$$

$$\vdots$$

$$-f_{m-1}^1 + A_{m-1}\bar{x}^1 + B_{m-1}u^1 = -f_m + A_m\bar{x} + B_mu$$

Noting the similarity between the terms on the night above, and the original definition set, we can make a judicious choice for f_i.

$$f_1 = y^1 - Du^1$$

$$f_2 = f_1^1 - B_1u^1 = y^2 - Du^2 - B_1u^1$$

$$\vdots$$

$$f_m = f_{m-1}^1 - B_{m-1}u^1 = y^m - Du^m - B_1u^{m-1} - \ldots - B_{m-1}u^1$$

which simplifies the form of equation set B to

$$\begin{bmatrix} C \\ A_1 \\ \vdots \\ A_{m-1} \end{bmatrix} \bar{x}^1 = A\bar{x} + Bu$$

Further choosing

$$\begin{bmatrix} C \\ A1 \\ \vdots \\ A_{m-1} \end{bmatrix} = I = \begin{bmatrix} 1 & & & \\ & 1 & & 0 \\ & & \ddots & \\ 0 & & & 1 \end{bmatrix}$$

Simplifies the form for the definition set

$$y = C\bar{x} + Du$$

$$\bar{x}^1 = A\bar{x} + Bu.$$

If this system represents a continuous system (differential equation), we have the familiar forms:

$$y(t) = C\bar{x}(t) + Du(t)$$

$$\dot{\bar{x}}(t) = A\bar{x}(t) + Bu(t)$$

If the system is discretized (difference equation) we have

$$y(t-mT_s) = C\bar{x}(t-mT_s) + Du(t-mT_s)$$

$$\bar{x}(t-(m-1)T_s) = A\bar{x}(t-mT_s) + Bu(t-mT_s)$$

Shifting t = 0 forward by m sampling intervals yields the more customary forms.

$$y(t) = C\bar{x}(t) + Du(t)$$

$$\bar{x}(t+T_s) = Ax(t) + Bu(t)$$

Knowing these forms the definition equations may be rewritten,

$$x_1 = y - Du$$

$$x_2 = x_1^1 - B_1u = y^1 - Du^1 - B_1u$$

$$\vdots$$

$$x_m = x^1_{m-1} - B_{m-1}u = y^{m-1} - Du^{m-1} - B_1u^{m-2} - \ldots - B_{m-1}u$$

The m^{th} row of A and the constant D, remain to be determined. Using the above expressions,

$$x^1_m = A_m \bar{x} + B_m u$$

expands to

$$\left.\begin{array}{l} A_{mi} (y - Du) + A_{m2} (y^1 - Du^1 - B_i u) + \ldots \\ + A_{mm} (y^{m-1} - Du^{m-1} - B_1 u^{m-2} \ldots - B_{m-1} u) - y^m + Du^m \\ + B_1 u^{m-1} + \ldots + B_{m-1} u^1 + Bm\, u \end{array}\right\} = 0$$

Associating like order terms.

$$\left.\begin{array}{l} A_{m1} y - (A_{m1} D + A_{m2} B_1 + \ldots + A_{mm} B_{m-1} - B_m) u \\ + A_{m2} y^1 - (A_{m2} D + A_{m3} B_1 + \ldots + A_{mm} B_{m-2} - B_{m-1}) u \\ \vdots \\ + A_{mm} y^{m-1} - (A_{mm} D - B_1) u^{m-1} \\ - y^m + D u^m \end{array}\right\} = 0$$

Which is very similar to equation 1 and may be set equal by allowing

$$D = b_0, \quad A_m = (- a_m, - a_{m-1}, \ldots, - a_1)$$

$$b_m = (A_{m1} D + A_{m2} B_1 + \ldots + A_{mm} B_{m-1} - B_m)$$

$$\vdots$$

$$b_1 = (A_{mm} D - B_1)$$

We now have arrived at the most common state space formulation. *

$$D = b_0 \qquad C = (1,\ 0,\ \dots\ 0)$$

$$A = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ -a_m & -a_{m-1} & \dots & \dots & -a_1 \end{bmatrix} \qquad B = \begin{bmatrix} B_1 \\ B_2 \\ \cdot \\ \cdot \\ \cdot \\ B_m \end{bmatrix}$$

where.

$$B_1 = b_1 - a_1 b_0$$

$$B_2 = b_2 - a_2 b_0 - a_1 B_1$$

$$\vdots$$

$$B_m = b_m - a_m b_0 - a_{m-1} B_{m-1}$$

and,

$$x_1 = y - b_0 u$$

$$x_2 = y^{1} - b_0 u^{1} - B_1 u$$

$$\vdots$$

$$x_m = y^{m-1} - b_0 u^{m-1} - B_1 u^{m-2} \dots - B_{m-1} u$$

AN ALTERNATE STATE SPACE FORM

Although in common use this form falls short of the goal of achieving maximum simplicity. To see why substitute $x_1 = y - b_0 u$ into equation 1 to obtain

$$\left.\begin{array}{l} a_m x_1 + (a_m b_0 - b_m)\, u + a_{m-1}\, x_1^{1} + (a_{m-1}\, b_0 - b_{m-1})\, u^{1} \\ + \ \dots\ + a_1\, x_1^{m-1} + (a_1 b_0 - b_1)\, u^{m-1} + x_1^{m} \end{array}\right\} = 0$$

We have defined a series of functions f_i so that the effects of the orders of all state variables in this equation are separated from the effects of the orders of u.

$$\bar{F} = A\bar{x} + Bu$$

in doing so the coefficients of the orders of x_1 above appear directly as entries of A. It is therefore reasonable to believe that if those definitions were chosen slightly

*See ref. 3.

differently, the form of B could also be simplified to express directly the coefficients of the orders of u above. That is.

$$B_i = (a_i b_0 - b_i)$$

We may maintain the correspondence of the rows of the definition set with the equations of set A, while obtaining the desired form for B by collecting all higher order terms together as a single first order state variable.

$$a_m x_1 + (a_m b_0 - b_m) u + x^1_k = 0$$

where

$$x^1_k = a_{m-1} x^1_1 + (a_{m-1} b_0 - b_{m-1}) u^1 + \ldots + x_1^m$$

decrementing the superscripts

$$x_k = a_{m-1} x_1 + (a_{m-1} b_0 - b_{m-1}) u + \underbrace{a_{m-2} x^1_1 + (a_{m-2} b_0 - b_{m-2}) u^1 + \ldots + x_1^{m-1}}_{x^1_{k-1}}$$

We now may continue defining the state variables in this manner until finally we get

$$x_{k-(m-2)} = a_1 x_1 + (a_1 b_0 - b_1) u + x^1_1$$

We now have a system of m + 1 equations each of which was obtained by decrementing the superscripts of the previous one. Therefore, to maintain the ordered correspondence with the lines of set A which were obtained by incrementing the superscripts, we must reverse the order of these equations setting $k = m$ so that

$$x^1_1 = - a_1 x_1 + x_2 + (b_1 - a_1 b_0) u$$

$$x^1_2 = - a_2 x_1 + x_3 + (b_2 - a_2 b_0) u$$

$$\vdots$$

$$x^1_{m-1} = - a_{m-1} x_1 + x_m + (b_{m-1} - a_{m-1} b_0) u$$

$$x^1_m = - a_m x_1 + (b_m - a_m b_0) u$$

Since the latest form of equation 1 on page B-8, is predicated on

$$y = x_1 + b_0 u$$

The values of $C = (1, 0 \ldots 0)$, $D = b_0$ are unchanged while the A matrix is altered slightly and the B vector is simplified.

$$A = \begin{bmatrix} -a_1 & 1 & 0 & \ldots & 0 \\ -a_2 & 0 & 1 & \ldots & 0 \\ \vdots & & & \ddots & \\ \vdots & & & & \cdot 1 \\ -a_m & 0 & 0 & \ldots & 0 \end{bmatrix} \qquad B = \begin{bmatrix} b_1 - a_1 b_0 \\ b_2 - a_2 b_0 \\ \vdots \\ \vdots \\ b_m - a_m b_0 \end{bmatrix}$$

The substitutions of page B-1 may be made for discrete systems.

MULTIPLE INPUT AND OUTPUT SYSTEMS

Until now, applications of state space notation only to single input, single output systems have been considered. The concept can now be extended to a network of linear elements. First consider a system of ℓ uncoupled linear elements each having its own set of state space equations of the form,

$$y_i = x_{1i} + b_{0i} u_i$$

$$\bar{x}_i^1 = A_i \bar{x}_i + B_i u_i$$

Combining all ℓ such sets into a single pair of matrix equations gives

$$\bar{y} = \bar{x}_1 + b_0 \bar{u}$$

$$X^1 = A^* X + B \bar{u}$$

where the dimensions of the equations have been expanded so that

$$\bar{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_\ell \end{bmatrix} \qquad \bar{u} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_\ell \end{bmatrix} \qquad \bar{x}_1 = \begin{bmatrix} x_{11} \\ x_{12} \\ \vdots \\ x_{1\ell} \end{bmatrix}$$

$$b_0 = \begin{bmatrix} b_{01} & & & 0 \\ & b_{02} & & \\ & & \ddots & \\ 0 & & & b_{o\ell} \end{bmatrix}$$

$$X = \underbrace{\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1\ell} \\ x_{21} & x_{22} & \cdots & x_{2\ell} \\ \vdots & \vdots & & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{m\ell} \end{bmatrix}}_{\ell} \Big\updownarrow m \qquad B = \underbrace{\begin{bmatrix} B_{11} & B_{12} & \cdots & B_{1\ell} \\ B_{21} & B_{22} & & B_{2\ell} \\ \vdots & \vdots & & \vdots \\ B_{m1} & B_{m2} & \cdots & B_{m\ell} \end{bmatrix}}_{\ell} \Big\updownarrow m$$

m = order of highest order element.

The columns of X are the state vectors of the ℓ elements and the columns of B are the corresponding input vectors. The matrix A^* always has a constant form, depending on the expressions chosen for the B matrix entries, of either

$$A^* = \begin{bmatrix} -a_{1i} & 1 & 0 & \cdots & 0 \\ -a_{2i} & 0 & 1 & \cdots & 0 \\ \vdots & & & \ddots & 1 \\ -a_{mi} & 0 & 0 & \cdots & 0 \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & \ddots & 1 \\ -a_{mi} & \cdots & \cdots & \cdots & -a_{1i} \end{bmatrix}$$

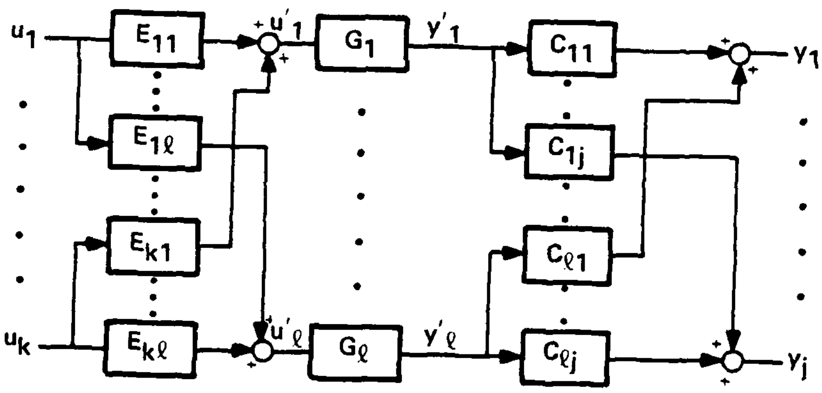
The multiplication A^*X is carried out in the usual row by column fashion except that a_{ji} is selected from the i^{th} row of the matrix

$$\begin{array}{cc}\begin{bmatrix} a_{11} & a_{21} & \cdots & a_{m1} \\ a_{12} & a_{22} & \cdots & a_{m2} \\ \vdots & \vdots & & \vdots \\ a_{1\ell} & a_{2} & \cdots & a_{m\ell} \end{bmatrix} & \updownarrow \ell \\ \longleftarrow m \longrightarrow & \end{array}$$

when multiplying any row of A^* by the i^{th} column of X.

Consider now the case in which the ℓ linear elements are coupled by pure gains. The inputs to the elements, $\bar{u}'$, may be expressed as ℓ linear combinations of k system inputs, $\bar{u}$.

SIMPLY COUPLED INPUT AND OUTPUT



That is

$$\bar{u}' = \begin{array}{cc} \ell \updownarrow & \begin{bmatrix} E_{11} & E_{21} & \cdots & E_{k1} \\ E_{12} & E_{22} & \cdots & E_{k2} \\ \vdots & \vdots & & \vdots \\ E_{1\ell} & E_{2\ell} & \cdots & E_{k\ell} \end{bmatrix} \\ & \longleftarrow k \longrightarrow \end{array} \bar{u}$$

The output may be similarly coupled giving (for j system outputs, y)

$$\bar{y} = \begin{array}{c} \\ j \\ \\ \end{array} \begin{array}{c} \begin{bmatrix} C_{11} & C_{21} & \cdots & C_{\ell 1} \\ C_{12} & C_{22} & \cdots & C_{\ell 2} \\ \vdots & \vdots & & \vdots \\ C_{1j} & C_{2j} & \cdots & C_{\ell j} \end{bmatrix} \\ \ell \end{array} \bar{y}'$$

Substituting into the multiple element state space equations

$$\bar{y} = C\bar{x}_1 + D\bar{u}$$

$$X^1 = A^* X + B\bar{u}$$

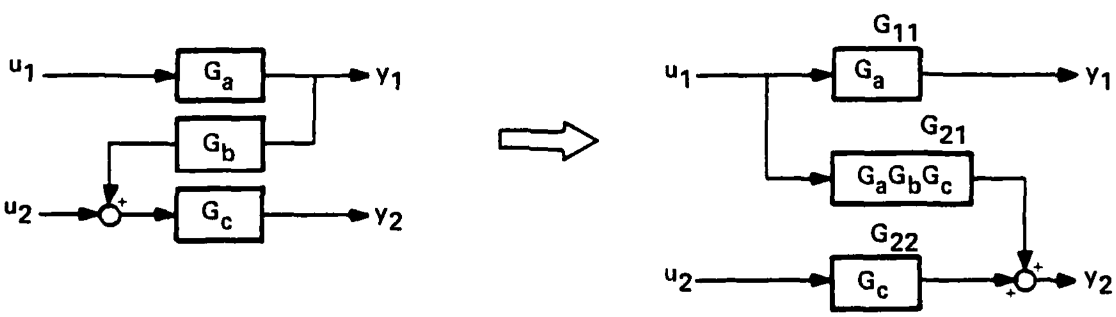
where

$$D = C b_0 E$$

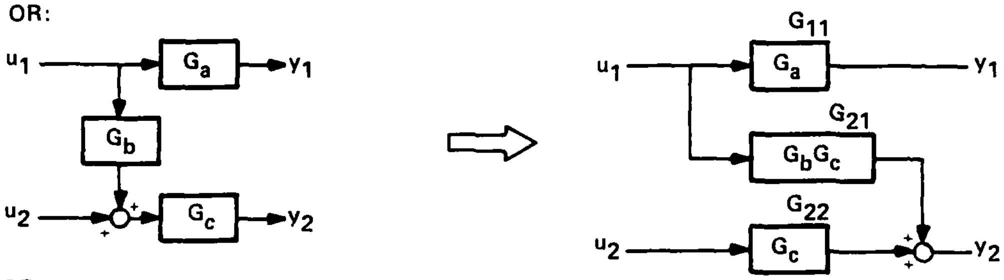
$$B = B' E$$

and B' is the B matirx on pg. B-11.

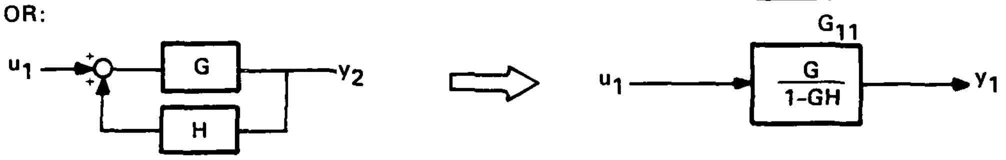
For more complex types of coupling, block diagram algebra may be used to reduce the network to this simple form. Some simple examples are:



OR:



OR:



It is clear that each coupling element, including pure gains, may be thought of as *creating a separate entry in the transfer matrix.*

$$\bar{y} = j \updownarrow \underbrace{\begin{bmatrix} G_{11} & G_{12} \cdots & G_{1k} \\ G_{21} & G_{22} \cdots & G_{2k} \\ \vdots & \vdots & \vdots \\ G_{j1} & G_{j2} \cdots & G_{jk} \end{bmatrix}}_{\longleftarrow\; k \;\longrightarrow} \bar{u}$$

This means that any network may be reduced so that entries of matrices C and E are either 0 or 1. In the case of matrix E, only one entry in each row may be nonzero. Furthermore, which entries are nonzero may be keyed to the subscripts of the transfer function; i. e. its location in the transfer matrix.

If $G_\ell = G_{jk}$, $E_{k\ell} = 1$ otherwise $E_{i\ell} = 0$

$C_{\ell_j} = 1$ $C_{\ell i} = 0$

All that is required to specify a highly general system by the method presented here is to specify the number of elements, their location in the transfer matrix, and the gains. poles, and zeroes of each element. The state spare matrices A^*, P, C and D may be calculated directly.

AD-A104 901 NAVAL AIR DEVELOPMENT CENTER WARMINSTER PA AIRCRAFT --ETC F/G 20/4
MATH MODEL FOR NADC SPIN SIMULATOR AND F-14 HIGH ANGLE OF ATTAC--ETC(U)
OCT 80 J M STIFEL
UNCLASSIFIED NADC-80220-60-VOL-1 NL

2 of 2
AD A
104901

END
DATE
FILMED
11-81
DTIC

DISTRIBUTION LIST

REPORT NO. NADC-80220-60

VOL I

AIRTASK NO. A0595941/0012/0241/000003

WORK UNIT NO. 1F0960GA

	Copies
NAVAIR (AIR-950 D)	6
(2 for retention)	
(2 for AIR-320D)	
(2 for AIR-5301)	
NASA Langley Research Center Langley Virginia (Attn: W. E. Gilbert)	2
NASA Dryden Flight Research Center Edwards AFB, Calif. (Attn: J. Gera, E-EDC)	2
Assistant Secretary of the Navy Research & Development Washington, DC 20350 (Attn: L. V. Schmidt)	2
DTIC	12

DATE
ILMED
-8